Security and Privacy in Location-Aware Mobile Systems



Christian Vaas

Wolfson College

University of Oxford

A thesis submitted for the degree of

Doctor of Philosophy

Michaelmas 2018

Acknowledgements

Personal

I thank my parents for their unconditional support, both financially and morally, without which I would not have been able to pursue this dissertation.

I thank my partner Estelle Beguin for making room to discuss my academic problems despite her own academic obligations. But even more, for the much needed non-academic counterbalance, motivating words, and patience.

I thank my friends at Wolfson College, who made my whole stay at Oxford an amazing, priceless experience. In particular I would like to extend my gratitude to Jack Tan, Lin Chen, Paige Mustain, Nawamin Pinpathomrat, Aileen Tse, Sabin Sulzer, and Joshua Combs.

I also want to thank all the rowers of Wolfson College Boat Club with who I had the pleasure to spent time on and off the water. Rowing gave me something to hold on to when the academic life seemed rather challenging.

I thank my research group in the Robert Hooke Building. Especially my colleagues in room 101 and Jantje Silomon for making workdays interesting and fun.

I thank my collaborators Marc Roeschlin, Kasper Rasmussen, N. Asokan, Mika Juuti, Mohammad Khodaei, and especially Panos Papadimitrators who was kind enough to provide the equipment used during the VANET experiments for this thesis.

Lastly, I want to thank my supervisor Ivan Martinovic, who introduced me to the initial idea for this dissertation. He also provided the necessary contacts for the collaborations which made my D.Phil at Oxford possible.

Institutional

I thank the EPSRC, the CDT in Cyber Security, the Department of Computer Science at Oxford, and Wolfson College for their generous funding which enabled me to pursue this work.

I also thank David Hobbs and Maureen York, who have been of tremendous help.

Abstract

For centuries, methods to obtain location information were crucial for our economic success. While initially requiring expert training and cumbersome tools like astrolabes and compasses, advances in material and engineering sciences have brought the ability to determine one's position on Earth to our fingertips. Devices as small and mobile as smart phones can now acquire and process location information.

Fuelled by this new data source, social media companies immediately realised the potential to enhance their users' experience and embedded location-based features into their products. Simultaneously, platforms which allow users to use location information as triggers for automation tasks emerged.

Beyond these use cases for entertainment and convenience, recent advances in academic and industrial research have started to leverage location information for security and safety purposes. While the advantages of location-aware mobile systems to improve security are undisputed, methods to ensure the reliability of location information and minimise the impact on user privacy still require further research.

In this dissertation, we aim to extend the knowledge about security and privacy implications of location information. To achieve this, we first analyse the different paths a mobile system can come in contact with this type of information. Based on these insights, we identify threats associated with the acquisition and processing of location claims. We recognise that in many cases, the absolute location of mobile devices is not needed but rather their constellation is sufficient. For example, the proximity of two devices can aid to validate location claims before relying on them for security or safety critical applications. We propose to use the trajectory a device takes to approach a location as that location's fingerprint. To show the feasibility of this idea and its potential applications, we evaluate two scenarios: user authentication and collective awareness in future intelligent transportation systems.

Finally, we analyse the consequences of data sharing for user privacy. More specifically, we investigate Vehicular Ad-hoc Networks (VANETs) as an open network where location information is essential to ensure its safe operation but also vastly available to malicious actors. We develop solutions to improve passenger privacy while providing more fine grained accountability in these networks which are a cornerstone in the future of intelligent transportation systems.

Contents

List of Figures **xiii**

List of Tables **xix**

List of Abbreviations **xxi**

1 Introduction **1**
1.1 Motivation 1
1.2 Published Work 4
1.2.1 Conference Papers 4
1.2.2 Posters and Demos 5
1.3 Work Done in Collaboration 6
1.4 Outline 7

2 Location Information in Mobile Systems: Threats and Solutions **9**
2.1 Acquisition 10
2.1.1 Dead Reckoning 10
2.1.2 Global Positioning System 11
2.1.3 Wireless Connectivity 12
2.2 Threats 13
2.2.1 Relay Attack 13
2.2.2 Spoofing Attack 15
2.3 Proposed Solutions 16
2.3.1 Distance Bounding 16
2.3.2 Ambient Context Comparison 18
2.3.3 Trajectory Recognition 19
2.4 Summary 20

3 Prover-verifier Proximity in Zero-Interaction Authentication Protocols **23**
3.1 User Authentication 24
3.2 System Proposal 25
3.3 Concept and Assumptions 27

3.3.1 CHAP Integration . . . 27
3.3.2 Application Scenario: Premise Access Control . . . 28
3.3.3 Adversary Model . . . 29
3.3.4 Design Goals . . . 29
3.4 STASH Architecture . . . 30
3.4.1 Trajectory Representation . . . 30
3.4.2 Primitive Generation . . . 32
3.4.3 Trajectory Comparison . . . 34
3.4.4 Comparison Thresholds . . . 36
3.5 Performance Evaluation . . . 38
3.5.1 STASH Prototype . . . 39
3.5.2 Movement Recognition . . . 41
3.5.3 Experimental Data Acquisition . . . 41
3.5.4 Path Separability . . . 41
3.5.5 Determining Initial Thresholds . . . 43
3.5.6 Impact of Reference Path Length . . . 45
3.5.7 Multi Path Learning . . . 46
3.5.8 Energy Consumption . . . 48
3.6 Security Analysis . . . 51
3.7 Discussion . . . 53
3.8 Summary . . . 54

4 Security and Privacy in Vehicular Ad-hoc Networks 57
4.1 Introduction . . . 58
4.2 Vehicular Communication . . . 59
4.3 Applications . . . 60
4.3.1 Safety . . . 61
4.3.2 Efficiency . . . 62
4.4 Requirements . . . 63
4.4.1 Authenticity . . . 64
4.4.2 Integrity . . . 64
4.4.3 Non-repudiation . . . 64
4.4.4 Conditional Anonymity . . . 65
4.4.5 Revocation . . . 65
4.5 Message Security . . . 65
4.5.1 Identity-based Cryptography . . . 66
4.5.2 Group Signatures . . . 66
4.5.3 Symmetric Cryptography . . . 67
4.5.4 Asymmetric Cryptography . . . 67

4.6 Passenger Privacy . . . 68
4.7 Pseudonym Change Schemes . . . 71
4.7.1 Silent Periods . . . 72
4.7.2 Encrypted Mix-Zones . . . 73
4.8 Summary . . . 74

5 Co-presence Verification in Vehicle Formations Using Driving Trajectories 75
5.1 Continuous Co-presence in Vehicle Formations . . . 76
5.2 System Proposal . . . 77
5.3 Concept and Assumptions . . . 79
5.3.1 Application Scenario: Formation Co-presence . . . 79
5.3.2 System Model . . . 81
5.3.3 Adversary Model . . . 83
5.3.4 Design Goals . . . 84
5.4 Get-In-Line Architecture . . . 86
5.4.1 Trajectory Representation . . . 86
5.4.2 Intended Trajectory . . . 87
5.4.3 Past Trajectory . . . 89
5.4.4 Co-presence Verification Protocol . . . 91
5.5 Evaluation . . . 95
5.5.1 Map Analysis . . . 96
5.5.2 Security Analysis . . . 101
5.5.3 Experiments . . . 106
5.6 Discussion . . . 109
5.7 Summary . . . 110

6 Cryptographic Mix-Zones for Private Pseudonym Change 113
6.1 System Proposal . . . 114
6.2 Concept and Assumptions . . . 116
6.2.1 Application Scenario: Chaff-based CMIX . . . 116
6.2.2 System Model . . . 118
6.2.3 Adversary Model . . . 119
6.3 Chaff-based Cryptographic mix-zone (CMIX) Scheme . . . 120
6.3.1 Credential Acquisition . . . 120
6.3.2 Chaff Notification . . . 121
6.3.3 Mix-zone Participation . . . 122
6.3.4 Cuckoo Filter Update . . . 123
6.4 Security Analysis . . . 124
6.4.1 Chaff Notifications . . . 124

6.4.2 Mix-zone Interaction . . . 124
6.4.3 Cuckoo Filter Update Propagation . . . 125
6.5 Privacy Analysis and Evaluation . . . 125
6.5.1 Exposure Metric . . . 125
6.5.2 Quantitative Analysis . . . 127
6.6 Performance Evaluation . . . 130
6.7 Physical Layer Tracking . . . 133
6.7.1 Methodology . . . 133
6.7.2 Experimental Setup . . . 133
6.7.3 Data Processing . . . 135
6.7.4 Results . . . 136
6.8 Discussion . . . 136
6.9 Summary . . . 137

7 Anonymous Driver-centric Message Authentication 139
7.1 Driver-centric Message Authentication . . . 141
7.2 System Proposal . . . 142
7.3 Concept and Assumptions . . . 143
7.3.1 Application Scenario: Strong Driver Authentication . . . 143
7.3.2 System Model . . . 145
7.3.3 Adversary Model . . . 146
7.4 Message Authentication Protocol . . . 147
7.4.1 Underlying Principle . . . 148
7.4.2 Enrolment . . . 150
7.4.3 Bionym Acquisition . . . 151
7.4.4 Message Authentication . . . 152
7.5 Security Analysis . . . 153
7.5.1 Passive Eavesdropper . . . 153
7.5.2 Bionym Linkability . . . 154
7.5.3 Active Manipulation . . . 154
7.5.4 Relay Attack Mitigation . . . 155
7.6 Performance Evaluation . . . 156
7.6.1 Biometric Performance . . . 156
7.6.2 Network Performance . . . 159
7.7 Discussion . . . 161
7.8 Summary . . . 162

8 Summary and Future Work 165
8.1 Summary of Results . . . 165
8.2 Future Work . . . 167
8.3 Final Conclusions . . . 169

Appendices

A Turning Angle Error **173**

References **175**

List of Figures

2.1 Taxonomy of ways to acquire absolute and relative location information in a mobile system. 10

2.2 Dead reckoning in aviation. Positions are indicated with timestamps, velocity and turning motion is used to estimate a plane's current location. 11

2.3 GPS is made up of 33 satellites orbiting the planet. 12

2.4 Connectivity based proximity detection. The wireless channel is commonly implemented using Bluetooth, Near-Field Communication (NFC), or WiFi. After a beacon indicates their presence, the two devices run a challenge-response authentication protocol. 13

2.5 An unintended communication channel is established between prover and verifier. *Ghost* and *leech* devices are connected through a fast, high-range communication channel. 14

2.6 Signal-based GPS spoofing attack against a drone in the presence of three satellites. The attacker's transmitter is indicated in red. . . . 15

2.7 Distance bounding protocol under relay attack. The adversary's transmissions are indicated in red, legitimate transmissions in black. 17

3.1 Architecture of a wireless ZIA scheme between a prover P and verifier V. The CHAP is shown in black while the components required to transform it to a zero-interaction protocol are indicated in purple. The proximity verification component introduced in our approach is shown in red. 28

3.2 P's approach trajectory can be described by a set of primitives needed to reach V. In this example, the primitives are: *move* 2, *left* 90°, *move* 3, *right* 30°, *move* 1, *left* 90°, *move* 1, *right* 30°, *move* 7, *left* 90°, *move* 2. 31

3.3 The data flow of the movement capturing system. The two Inertial MEMS sensors and a third software sensor (gravity) provide the data for the movement and turn detection algorithms. These produce movement M, stationary S, and turns L, R primitives. 32

3.4 Prover-side proximity verification: The prover compares the current candidate path against reference paths in the repository. It only computes the response to the verifier's challenge if the paths match. Otherwise the protocol is aborted unless the user explicitly approves the authentication run. 35
3.5 The confidence factor λ increases with the number of observed instances of the same reference path. For every two new observations, its distance to one is cut in half, i.e. the confidence grows logarithmically. 38
3.6 Android prototype of STASH: (a) Movement is classified in one second intervals. A button allows the user to lock the verifier in case of a false accept. (b) Interface to approve authentication attempts of unrecognised paths. The temporal length of the reference path can be adjusted using a slider at the top of the screen. The blue and green circle indicate the start and finish of the current trajectory. . 40
3.7 An example of within-class (blue) and between-class similarities (red) for one reference path. Choosing a reference path-specific threshold of 18 minimises EER on this route. 42
3.8 False accept and false reject error rates depending on the value chosen for α. For each α, the initial threshold that yields the lowest combined error is given. Large α values result in low FRR, while small values result in low FAR. 44
3.9 Based on the length specific initial threshold values, we compute the error rates FAR (red) and FRR (blue) for 20 routes (20 lines). The testing set for each route consists of paths of all other routes. . . . 45
3.10 Coordinates denote median (FAR, FRR), numbers describe number of instances used. Increasing the number of observed reference path instances leads to a decrease in FARs and FRRs. The extent of this effect varies depending on the used threshold derivation algorithm. 47
3.11 **Mixed decision thresholds**. Box-and-whiskers plot over FARs and FRRs of different temporal lengths after STASH has observed five instances of a reference path. On average FARs are below 3% and FRRs are below 0.5% if $L \geq 2min$. 49

4.1 The entities in a VANET are RSUs and OBUs. Using the WiFi-like IEEE 802.11p standard, location based information is shared via V2V and V2I communication. 60
4.2 Certification hierarchy of the VPKI. Solid black arrows indicate trust relations, dashed lines show pseudonym requests, and red dotted lines highlight pseudonym acquisition. 69

4.3 Two vehicles with different pseudonym strategies. While OBU_0 signs messages with the same pseudonym for its entire journey, OBU_1 uses three pseudonyms to separate its trip into segments. Transitions between pseudonyms are highlighted with red dashed circles. . . . 72

5.1 A vehicle formation with an LV, two FVs and a CV. At time t_0, the CV senses the group and decides to follow it. Next, at t_1 the LV and the CV shared the same trajectory which allowed the CV to become a following vehicle. Like the other FVs it now continuously verifies its co-presence using beacon messages that include its past trajectory. After t_2, a follower leaves the group, the lack of beacon messages between the LV and the FV terminates their sessions making it a CV. 80

5.2 Map data and IMU measurements are used to generate intended and past trajectories. Their privacy preserving comparison allows two vehicles to first determine whether their future routes are going to overlap and to establish a symmetric key to secure further communication. After a CV has been verified and thus been promoted to a FV, their ongoing co-presence is continuously verified. 87

5.3 Map segment with an overlay of the road network. The LV and CV are currently driving towards v_0. Vertices and edges in red outline their future route used for intended trajectory comparison. 89

5.4 Gyroscope signal with peaks indicating turns where red denote left and green denote right turns. Orange areas were classified as inter-turn travel duration. 91

5.5 Phase 1 of the protocol privately compares the intended trajectories Seq_{CV} and Seq_{LV} for a given length d_{in}. Participants only learn about an exact match, which allows the derivation of the shared value, $seed_0$. 94

5.6 We base our map analysis on the road networks of four cities: Berlin, Singapore, San Francisco, and Oxford. By choosing different geographical regions (North America, Europe, and Asia), construction epochs, and architectural styles we aim to analyse the effect of these road layouts on the security and performance of our system. 97

5.7 Example of a road segment with heading change α for a turn around V_2 from V_1 towards V_3. We use the Cosine-Haversine formula by Robusto [164] to compute the length of an edge using the GPS coordinates of its two vertices. 98

5.8 Turning angle distributions for the four cities. The absolute numbers of turns are given for each angle value from 0° to 180°. 98

5.9 A vehicle moves straight with 50% probability, and turns left or right with 25% probability respectively. It is allowed to turn around if no other choice exists. Angles between 30° and 150° are turns while any smaller angles are forward and larger angles are considered backward movement. 99
5.10 Trajectory feature impact: Number of turns included and overall length of a trajectory. (a) Average number of collisions per trajectory for each city from 9000 randomly chosen trajectories with three, six, and nine turns per trajectory. (b) Average number of collisions per trajectory for each city from 9000 randomly chosen trajectories of 9 different lengths. 100
5.11 Search space size of intended trajectories for a location unaware attacker with respect to d_{in}. Average number of possibly matching trajectories in each city for $d_{in} \in [10, 50, 100, 150, 200]$ meters. . . . 101
5.12 False accept rates in phase 1, intended trajectory matching, for all four cities when an attacker attempts to predict d_{in} meters of the victim's future route. 103
5.13 (a) Comparison of the two approaches, trajectory information significantly decrease the FAR for a given length allowing quicker verification of a CV and detection of a no longer co-present FV. (b) Trajectory beacon approach simulated in all four cities. 105
5.14 Combined false accept rates for phase 1 and 2 for all four cities with respect to distance since phase 1 initiation. 106
5.15 Experimental route with a overall length of 5.5 km, selected for the analysis of phase 2. An impasse at A serves as start and endpoint. 107
5.16 Experimental setup for the two vehicles scenario: (a) A 2016 Ford Focus (right) and a 2017 Vauxhall Astra (left) which were used as LV and CV (b) Smart phones were positioned in the centre console of each car. 107
5.17 Gyroscope readings of LV and CV from an experiment run. Errors are introduced by differences in amplitude or displaced peaks. . . . 108

6.1 Network of four mix-zones: A, B, C, and D. Their encrypted regions are indicated with dotted blue lines. Green and orange solid lines show two real vehicles; each dashed blue line denotes the trace of a chaff vehicle. 117
6.2 RSU r notifies its adjacent RSUs o about the fingerprint of a chaff pseudonym. The RSUs reply with m indicating how many vehicles are in transit towards r since the last notification. 121

6.3 Mix-zones are advertised through periodic beacons. When approaching a mix-zone, OBUs prepare a new key pair, provide their intended trajectory, and obtain the symmetric mix-zone key. 122
6.4 OBUs forward CF updates and mix-zone keys epidemically outside RSU coverage. 123
6.5 The areas used for the simulation expose individual mobility patterns due to their driver population and road layout. Road capacities are colour coded, yellow points show junctions while dashed circles indicate encrypted mix-zones with $R^{MIX} = 50$m. 126
6.6 Privacy degradation comparison: lines show exposure values for the 90^{th} percentile of the vehicle population. 128
6.7 Effect of mix-zone encryption range on vehicle exposure. The attackers observe 30% and 100% of the mix-zones. 129
6.8 Active chaff vehicles per real vehicle. The shaded area denotes number of real vehicles. Each line represents one mix-zone: 28, 31, 18 for the central, suburban, and residential region. 131
6.9 Vehicle with NEXCOM OBU (a) & (b) and two MIMO synchronised Ettus N210 USRPs used to acquire I/Q samples (c). 134
6.10 Our attack scenario; each road of the intersection is assigned to a sector (A, B, and C). A blue cross indicates the location of a received packet. The stationary attacker is shown with a red cross. 135

7.1 Data flow: A driver requests bionyms by providing her biometric measurement. The bionyms are anonymised signing material that is used to generate VANET message signatures. 144
7.2 System model with the three VANET entities: TA, OBUs, and RSUs. RSUs relay information between OBUs and the TA. Vehicles can acquire a driver's characteristics via sensors embedded into the steering wheel. 145
7.3 Enrolment of driver D at the TA. After her identity is confirmed, the enrolment commitment C_D is established and stored to verify D during bionym acquisitions. 150
7.4 Acquisition of N bionyms: A driver provides her biometric measurement before OBU v requests and stores a batch of bionyms from the TA. 151
7.5 Authentication of message m: OBU v signs m and transmits the message, signature, and bionym over the network. Nearby vehicles $v^\star$ validate the bionym and verify the signature. 152

7.6 Measurement set-up: A waveform generator produces the frequency sweep which is received and recorded by an oscilloscope. When the human subject completes the electric circuit, the body impedance Z_{human} can be acquired. 157
7.7 Body impedance recognition performance for 5 different metrics (a) and receiver operating curve (b). Results are averaged over the test population and obtained through 5-fold cross-validation. Bars represent 95% confidence intervals. 159
7.8 Average revocation delay for three different bionym lifetimes relative to trip length. A value of 1.0 means revocation after current trip. Error bars represent 95% percentiles across the vehicle population. 160

A.1 Absolute measurement error of angles for left an right turns captured at 20 Hz using two phones placed in the experimental vehicle. . . . 173

List of Tables

3.1 Mean and standard deviation of FAR/FRR using only the initial decision threshold the 20 paths in our dataset. 46
3.2 Net increase of hourly battery consumption in percentage of total capacity and milliampere in the office worker scenario. 50
3.3 Net increase of hourly battery consumption in percentage of total capacity and milliampere in the always awake scenario. 50

5.1 Pre-computed and public parameters of the private trajectory comparison protocol. 93
5.2 Summary of the experiment parameters and resulting error rates averaged across all runs. 108

6.1 Protocol parameters summary and description. 120
6.2 Metric parameters summary and description. 127
6.3 Key figures for the simulated regions ($R^{MIX} = 50$ m) 127
6.4 Different encryption radii affect the number of simultaneously active chaff pseudonyms, chaff message generation, and pseudonym pool size. 130

7.1 Summary of protocol parameters and notations. 148

List of Abbreviations

AES Advanced Encryption Standard

BSM Basic Safety Message

CA Certificate Authority

CAM Cooperative Awareness Message

CACC Cooperative Adaptive Cruise Control

CCA Cooperative Collision Avoidance

CF Cuckoo Filter

CHAP Challenge-Handshake Authentication Protocol

CMIX Cryptographic mix-zone

CRL Certificate Revocation List

CV Candidate Vehicle

DSRC Dedicated Short Range Communication

DoS Denial of Service

ECC Elliptic Curve Cryptography

ECDSA Elliptic Curve Digital Signature Algorithm

ECG Electrocardiography

EEBL Electronic Emergency Brake Light

EER Equal Error Rate

FAR False Accept Rate

FRR False Reject Rate

FV Follower Vehicle

GLONASS GLObal NAvigation Satellite System

GPS Global Positioning System

HMAC Hashed-Message Authentication Code

HMM Hidden Markov Model

HSM Hardware Security Module

I/Q In-phase and Quadrature

IBC Identity-Based Cryptography

IMU Inertial Measurement Unit

IP Internet Protocol

ITS Intelligent Transportation System

LDW Local Danger Warning

LR Logistic Regression

LTC Long Term Certificate

LTCA Long Term CA

LTE Long-Term Evolution

LV Leader Vehicle

LuST Luxembourg SUMO Traffic

MANET Mobile Ad-hoc Network

MAC Media Access Control

MEMS Micro-ElectroMechanical System

MIMO Multiple-Input and Multiple-Output

NFC Near-Field Communication

NW Needleman-Wunsch

OBU On-Board Unit

OSM OpenStreetMap

PCA Pseudonym CA

PKI Public-Key Infrastructure

RA Resolution Authority

RCA Root CA

RSSI Received Signal Strength Indication

RSU Road-Side Unit

SDR Software Defined Radio

SUMO Simulation of Urban MObility

TA Trusted Authority

TLS Transport Layer Security

TaaS Transportation as a Service

ToF Time-Of-Flight

UAV Unmanned Aerial Vehicle

USRP Universal Software Radio Peripheral

V2I Vehicle-to-Infrastructure

V2V Vehicle-to-Vehicle

V2X Vehicle-to-X

VANET Vehicular Ad-hoc Network

VPKI Vehicular Public-Key Infrastructure

VTL Virtual Traffic Light

XOR Exclusive-Or

ZIA Zero-Interaction Authentication

In the beginning it was all black and white.

— Maureen O'Hara

1 Introduction

Contents

1.1 Motivation **1**
1.2 Published Work **4**
1.2.1 Conference Papers 4
1.2.2 Posters and Demos 5
1.3 Work Done in Collaboration **6**
1.4 Outline **7**

1.1 Motivation

Guided by astronomic constellations, vessels were already manoeuvred hundreds of years ago using celestial navigation [1]. Since then, a series of ground breaking inventions, such as the astrolabe [2], the compass [3], and radionavigation-satellite services [4] have revolutionised the way we acquire and process location information. These new techniques allowed increasingly precise location determination, while at the same time advances in the material and engineering sciences enabled their miniaturisation and integration into mobile devices.

Today, radionavigation-satellite services are indispensable, with the first deployment being the American Global Positioning System (GPS). While initially reserved for military applications, its opening to civil use was followed by a rapid

adoption which lead to the ubiquitous availability of location information in smart phones and other customer products. Social media companies and advertisers were among the first to recognise the potential of location-based services. Platforms like Yelp, Foursquare, and Facebook allowed their users to perform location-based queries as well as provide friends with updates on their current location. Companies like Google, TomTom, and Garmin realised that personal navigation can be brought to the users' fingertips using smart devices.

Nowadays, the diversity of location-based applications has virtually exploded. In particular, convenience features that automate tedious tasks have been picked up by customers in no time. For example, apps like Tasker[1] and IFTTT[2] allow the user to trigger predefined actions, such as turning on the light when coming home or providing a reminder to stop when passing the supermarket.

In 2015 already nine-in-ten cellphone owners frequently carried their phone with them[3], making the phones' location a proxy for their owner's whereabouts. With the advent of smart watches, which effectively offer the same connectivity and sensing capabilities as smart phones, an increasing number of users are expected to always carry at least one smart device. Security systems have begun to leverage this trend to improve the protection of digital as well as cyber-physical assets. For example, banking start-ups like Revolut[4] and Monzo[5] use the location of their customers' smart phones as cues to validate transactions and combat credit card fraud.

While the aforementioned use cases require the user's absolute location, the constellation, i.e. the proximity, of the user to a device is often sufficient. One measure for proximity can be the distance of two absolute locations, e.g. between two GPS coordinates. Other approaches enable mobile devices to sense each other's presence and derive an estimate of their proximity. Independent of the way this information is obtained, a mobile device's proximity is increasingly utilised to improve the usability of security systems.

[1]Tasker for Android, total automation for Android (https://tasker.joaoapps.com/)
[2]If This Then That, applets for automation (https://ifttt.com/)
[3]http://www.pewinternet.org/2015/08/26/chapter-1-always-on-connectivity/
[4]https://www.revolut.com/
[5]https://monzo.com/

For instance, the dominating implementations of user authentication still rely on the use of passwords, a cumbersome method of authentication [5]. This has sparked the development of transparent authentication methods. These exploit cues based on user behaviour, biometrics, or the environmental context to make the authentication process effortless [6–8]. One instance of a transparent authentication scheme is Zero-Interaction Authentication (ZIA), which relies on a verifier to automatically authenticate a user when her prover device is in proximity [9]. In ZIA schemes a verifier typically ensures the proximity of the prover by measuring either the strength of radio signals over some short range wireless channel [10] or the time required for a transmission over that channel [11, 12]. However, these techniques are vulnerable to relay attacks [11, 13]. During such an attack, the attacker forwards messages between prover and verifier when they are not co-located, thus falsifying the proximity measure of the verifier.

Proximity information can also be used to verify location claims of third parties. For example, in an environment with many physical entities such as roads, a new collaborative approach to increase safety and efficiency is emerging. Vehicular Ad-hoc Networks (VANETs) are supposed to be integrated into the next generation of cars heralding the advent of intelligent transportation systems [14–17]. Connected vehicles are envisioned to exchange their location, heading, and velocity in order to provide a common understanding of their constellation, hence avoiding accidents and increasing road throughput. However, malicious network participants can provide false information in order to turn these benevolent applications against other road users. Such attacks can target safety applications like emergency braking and danger warning with consequences such as collisions or even fatalities [18–20].

Despite the clear relevance of location information for security, a user's location also bears the ability to subvert her privacy. While a compromise can happen rather obviously, for example by continuously sharing a user's location with her surroundings as it is done in VANETs [21, 22], other channels that can leak location information are not that apparent. A malicious smart phone application, for example, can use sensors such as gyroscope and accelerometer, which do not

require explicit user consent, to infer her current location whilst driving or using public transport [23–25].

Although correct location information provides great merit for users and can make our lives more convenient, location-aware mobile devices have to exercise care when relying on this data. We believe that the potential of location information is not yet fully exhausted and that a new view on the security and privacy implications of this data is needed to foster future innovation. Therefore, we formulated the following research goals for this thesis:

- Analyse location-based mobile systems, identify different attack vectors, and evaluate how proximity measures can be used to verify location information.
- Develop practical solutions that provide the guarantees required for a secure and safe operation of these systems.
- Understand the privacy implications of information location-aware mobile systems share with third parties.
- Devise schemes that improve user privacy in existing location dependent applications while not limiting their core functionality.

1.2 Published Work

During my time as a D.Phil candidate, the following pieces of work were created. They provide the foundation for this dissertation. We are going to give a detailed list of the respective contributions in each chapter and reference the relevant research artefacts.

1.2.1 Conference Papers

- Mika Juuti, Christian Vaas, Ivo Sluganovic, Hans Liljestrand, N. Asokan and Ivan Martinovic. STASH: Securing transparent authentication schemes using prover-side proximity verification. 14th Annual IEEE International Conference on Sensing, Communication, and Networking, SECON 2017, San

Diego, CA, USA, June 12-14, 2017.
Available: https://doi.org/10.1109/SAHCN.2017.7964922

- Christian Vaas, Mika Juuti, N. Asokan and Ivan Martinovic. Get in Line: Ongoing Co-presence Verification of a Vehicle Formation Based on Driving Trajectories. IEEE European Symposium on Security and Privacy, EuroS&P 2018, London, United Kingdom, April 24-26, 2018.
Available: https://doi.org/10.1109/EuroSP.2018.00022

- Christian Vaas, Mohammad Khodae, Panos Papadimitratos and Ivan Martinovic. Nowhere to hide? Mix-Zones for Private Pseudonym Change using Chaff Vehicles. 2018 IEEE Vehicular Networking Conference, VNC 2018, Taipei, Taiwan, December 5-7, 2018.
Available: https://ieeexplore.ieee.org/abstract/document/8628449

- Marc Roeschlin, Christian Vaas, Kasper Rasmussen and Ivan Martinovic. Bionyms: Driver-centric Message Authentication using Biometric Measurements. 2018 IEEE Vehicular Networking Conference, VNC 2018, Taipei, Taiwan, December 5-7, 2018.
Available: https://ieeexplore.ieee.org/abstract/document/8628359

1.2.2 Posters and Demos

- Mika Juuti, Christian Vaas, Ivo Sluganovic, Hans Liljestrand, N. Asokan and Ivan Martinovic. *Demo:* Implementing Prover-Side Proximity Verification for Strengthening Transparent Authentication. 14th Annual IEEE International Conference on Sensing, Communication, and Networking, SECON 2017, San Diego, CA, USA, June 12-14, 2017.
Available: https://doi.org/10.1109/SAHCN.2017.7964951

- Christian Vaas, Panos Papadimitratos and Ivan Martinovic. *Poster:* Increasing Mix-Zone Efficacy for Pseudonym Change in VANETs using Chaff Messages. Proceedings of the 11th ACM Conference on Security & Privacy in Wireless

and Mobile Networks, WiSec 2018, Stockholm, Sweden, June 18-20, 2018.
Available: https://doi.acm.org/10.1145/3212480.3226103

- Christian Vaas, Marc Roeschlin, Panos Papadimitratos and Ivan Martinovic. *Poster:* Tracking Vehicles Through Encrypted Mix-Zones Using Physical Layer Properties. 2018 IEEE Vehicular Networking Conference, VNC 2018, Taipei, Taiwan, December 5-7, 2018.
Available: https://ieeexplore.ieee.org/abstract/document/8628387

1.3 Work Done in Collaboration

My supervisor Prof. Ivan Martinovic togehter with Prof. N. Asokan, contributed the initial idea of using changes in WiFi signal strength in combination with inertial sensor readings for security applications.

I worked together with Mika Juuti, N. Asokan's PhD student, on the realisation of the prover-side proximity verification presented in Chapter 3. N. Asokan provided high-level feedback during my stay at Aalto University Helsinki. Mika wrote code to train and measure the machine learning models. Hans Liljestrand, a Master's student at Aalto University, recorded smart phone battery levels required to assess the power consumption of our prototype.

Mohammad Khodae, Panos Papadimitratos' PhD student, gave high-level feedback on the paper *Nowhere to hide?* which forms the basis for Chapter 6. During the experiments for the second half of that chapter, Marc Roeschlin, Prof. Kasper Rasmussen's D.Phil student, drove one of the experiment vehicles and helped to manage the data acquisition.

Marc Roeschlin also worked on and equally contributed to the research used in parts for Chapter 7. The machine learning on biometric data, which is mainly Marc's contribution, is not focus of this dissertation and hence only included for completeness. Kasper Rasmussen provided high level feedback for this work.

1.4 Outline

In this section, each chapter is briefly summarised to outline the structure of this dissertation:

- **Chapter 2** provides background information on the security and privacy implications of location data accessible to mobile systems. Based on this, we describe the threats systems face when relying on location information. We review the existing work concerned with these attacks to be able to compare them to our proposals in the later chapters.

- **Chapter 3** looks at proximity verification in zero-interaction authentication protocols. We present an approach to address relay attacks on such systems in a semi-mobile scenario. Our scheme enables a mobile prover to ensure its co-presence to a stationary verifier based on the user's approach trajectory. During our evaluation, we assess the system's security and usability based on data collected in two cities. To ensure deployability, we measure the energy consumption of our prototype implementation.

- **Chapter 4** provides the necessary background for the subsequent chapters by describing the security and privacy requirements of vehicle-to-vehicle communication in future VANETs. After reviewing the widely agreed architecture for message authentication, we give a more in-depth description of challenges when providing driver privacy. For this, we discuss the use of pseudonymous credentials and attacks against different pseudonym change strategies.

- **Chapter 5** proposes a system that determines the relevance of VANET message. Current VANET security standards guarantee the authenticity and integrity of exchanged messages, however data semantics reflecting physical properties like spatial proximity and driving direction are not considered. By extending the idea from Chapter 3, a mechanism to provide ongoing co-presence verification of two vehicles based on their driving trajectories is devised. Our system verifies location claims and prevents the injection of

non-existing ghost vehicles. We show the feasibility of our approach using real-world experiments, conceptually analyse our protocol, and perform simulations under realistic traffic conditions to evaluate system performance.

- **Chapter 6** shifts the focus from message security to driver privacy. We consider encrypted mix-zones as one instance of a pseudonym change strategy that aims to improve user privacy. Our analysis reveals dependencies between the efficacy of mix-zones and the driver population, road layout, and time of day, which has not been considered in previous work. Based on these findings, we present a scheme that works independent of vehicles' mobility patterns. To show the performance of our chaff-based approach, we conduct large scale simulations under realistic traffic conditions and compare the results to an existing cryptographic mix-zone scheme.

- **Chapter 7** introduces a new VANET message authentication scheme with the driver's digital identity at heart. Current schemes link an owner's identity to her vehicle's identity at registration time. However, if a vehicle is shared among individuals, the causer of an accident is not necessarily the owner of the vehicle and therefore might be difficult to determine. Our system provides the security guarantees required for VANETs and protects a driver's privacy while tying the identity of the driver to emitted messages. In a systematic security analysis we show the correctness of our protocols. Additionally, we conducted a user study under simulated driving conditions to demonstrate the feasibility of acquiring biometric measurements while driving a vehicle.

- **Chapter 8** summarises our findings, outlines the future work required to ensure the secure and privacy preserving operation of future intelligent transportation systems, and concludes this dissertation.

Dare to err and to dream. Deep meaning often lies in childish plays.

— Friedrich Schiller

2

Location Information in Mobile Systems: Threats and Solutions

Contents

2.1 Acquisition 10
2.1.1 Dead Reckoning 10
2.1.2 Global Positioning System 11
2.1.3 Wireless Connectivity 12
2.2 Threats 13
2.2.1 Relay Attack 13
2.2.2 Spoofing Attack 15
2.3 Proposed Solutions 16
2.3.1 Distance Bounding 16
2.3.2 Ambient Context Comparison 18
2.3.3 Trajectory Recognition 19
2.4 Summary 20

While location information is commonly understood as a point in three-dimensional space, many applications in fact do not require the absolute location of a device but only knowledge about its position relative to other devices. Prominent examples of this are: device pairing, contactless payments, and keyless-entry systems. For these use cases, confirming the proximity of two devices, e.g., between a smart phone and a payment terminal, or a car key and a vehicle, is sufficient for their secure application. The common denominator in these scenarios is that the closer two

devices are, the higher is the level of security as attacks or mix-ups become less likely. In other cases, however, protocols that can only be applied whilst in close proximity, might not be feasible and systems resort to the use of absolute location information. In VANETs for instance, awareness messages are tagged with GPS coordinates to indicate their relevance to other traffic participants.

In this chapter, we first describe how absolute and relative location information can enter a mobile system, then we review common threat models against such data, and finally survey the current literature which tries to address these.

2.1 Acquisition

Several techniques to acquire location information have been researched and reached real-world deployment. As shown in the taxonomy in Figure 2.1, we can categorise these according to the origin of data which can either be *internal* or *external* to the mobile device.

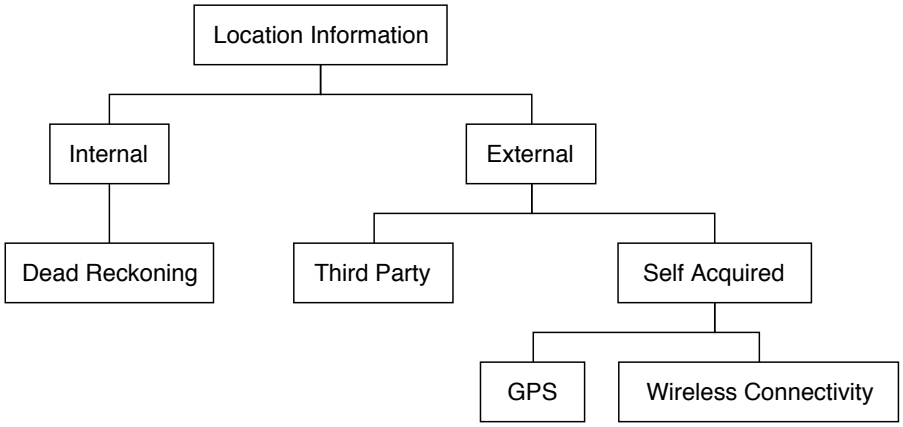


Figure 2.1: Taxonomy of ways to acquire absolute and relative location information in a mobile system.

While dead reckoning, a method known from marine navigation, relies on internally measured information, location data can also be provided by a third party or acquired using external reference signals.

2.1.1 Dead Reckoning

Before GPS became available, dead reckoning was the prevalent method for

marine and aerial navigation. As shown in Figure 2.2, outgoing from a known location A (indicated in green), local measurements such as velocity readings and turning angles were used to determine the current position of the plane [26]. While this method allows estimation of the vehicle position without external signals, it is also susceptible to errors. Common sources for location errors are inaccuracies introduced when determining the velocity or heading. These errors may accumulate until a new known position can be established through some external landmark. Advances in Inertial Measurement Unit (IMU) technology allowed for the integration of formerly large instruments like a sextants into modern mobile devices, such as smart phones. Using miniaturised gyroscopes and accelerometer, a substantial amount of research has centred on indoor localisation [27] and pedestrian dead reckoning [28]. Nowadays, inertia based positioning is still used when GPS signals are not available.

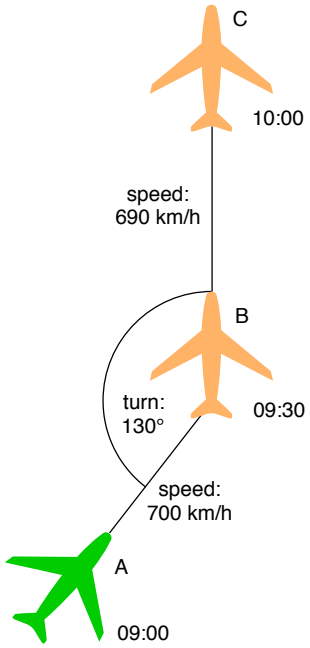


Figure 2.2: Dead reckoning in aviation. Positions are indicated with timestamps, velocity and turning motion is used to estimate a plane's current location.

2.1.2 Global Positioning System

Deployed by the United States in 1973, GPS is a radio-navigation system operated by the US Air Force, which was at first exclusively available for military use. To date, 33 satellites orbit our planet as shown in Figure 2.3[1]. To compute a device's location anywhere on Earth, the time of arrival of GPS signals is measured and used for triangulation. As long as the receiver maintains line of sight with at least four signal sources, sufficient data is available to acquire a so called GPS fix. The resulting GPS coordinates reflect the receiver's location within an error of 3-5 meters. While the most common error source, unsynchronised clocks, is continuously monitored and adjusted, GPS signals, like any other wireless transmissions, are susceptible to jamming.

[1]Source for Figure 2.3: https://www.gps.gov/

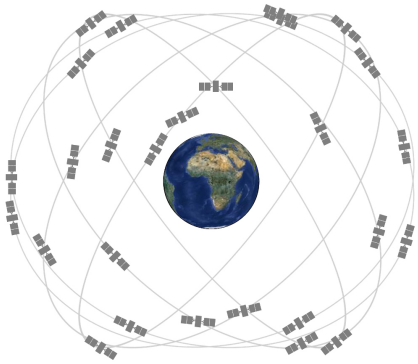


Figure 2.3: GPS is made up of 33 satellites orbiting the planet.

Furthermore, the signals emitted by GPS satellites are unauthenticated pseudorandom bit sequences which can be spoofed with a sufficiently powerful transmitter [29]. Although GPS was opened for civilian use in the 1980s, the fact that it is owned and controlled by the US has led to alternative deployments. The Russian GLObal NAvigation Satellite System (GLONASS), the European Galileo, and the Chinese BeiDou-2[2] are also satellite-based navigation systems designed to work independently of each other. Yet, since the underlying technology for all radio-navigation systems is the same, modern receivers have the ability to fuse location estimates from multiple sources to improve accuracy.

2.1.3 Wireless Connectivity

While the previous two approaches allow us to obtain absolute location information, it is sufficient for some applications to determine whether two devices are within a certain distance of each other. A common approach is the use of beacon messages over some wireless channel. Figure 2.4 illustrates the basic principle of such systems on the basis of a transparent authentication scenario. One of the two devices continuously announces its presence using beacon messages. When a potential partner device receives these messages, it either immediately assumes the proximity of the two devices [10, 30] or uses transmission channel information such as Received Signal Strength Indication (RSSI) to further determine their actual distance [31, 32]. In either case, the systems run the proximity-based protocol afterwards. Similar to GPS, the wireless nature of the exchanged signals makes them susceptible to jamming attacks. More importantly, even if the beacons are authenticated, an attacker could forward them to make the system believe that two distant devices are in proximity.

[2] http://en.beidou.gov.cn/

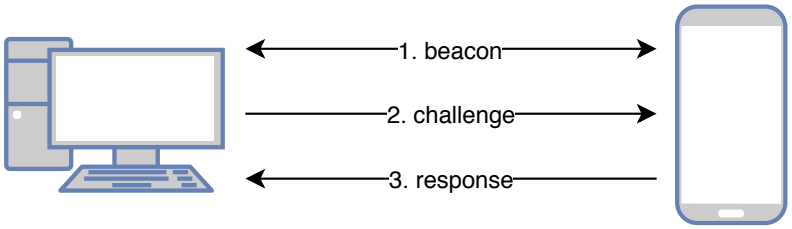


Figure 2.4: Connectivity based proximity detection. The wireless channel is commonly implemented using Bluetooth, NFC, or WiFi. After a beacon indicates their presence, the two devices run a challenge-response authentication protocol.

2.2 Threats

Based on the presented acquisition methods, we discuss threats faced by mobile devices when relying on location information. These are going to form the basis of the adversary models used to analyse the systems proposed throughout this dissertation.

2.2.1 Relay Attack

The attacker model first presented by Dolev and Yao [33] is widely used to describe the capabilities of wireless attackers. Adversaries under this model can overhear, intercept, and synthesise wireless messages but are not capable of breaking the confidentiality or integrity of a secure communication channel. Applied to wireless connectivity based proximity determination, such an attacker can be understood to be similar to a WiFi extender. The extender's purpose is to relay the Service Set Identifier (SSID) of the host WiFi and hence emulate its presence in places not covered by the original transceiver. The same technique can be used to subvert a proximity based authentication system.

Two devices, *ghost* and *leech* are required to conduct such a relay attack, also known as wireless man-in-the-middle attack, see Figure 2.5. Connected via a fast high-range communication channel, these can collaboratively establish a connection between the prover, e.g. a smart phone, and the verifier, e.g. a personal computer. The leech, placed in the vicinity of the prover, pretends to be the verifier, while the ghost, which resides in proximity of the verifier, impersonates the prover by bidirectionally relaying messages between them. As shown, a proximity based authentication system can be defeated through a relay attack. Even if the verifier

implements a distance check using RSSI, it will measure the value between itself and the ghost instead of the actual prover device.



Figure 2.5: An unintended communication channel is established between prover and verifier. *Ghost* and *leech* devices are connected through a fast, high-range communication channel.

The severity of relay attacks depends on the deployment scenario. In a system like PhoneAuth [31], which uses device proximity as a second factor, the attacker can reduce the security to the strength of the primary factor. However, schemes that only implement one authentication factor can be completely defeated. Examples include Hermes, BlueProxmity, and the scheme by Choi *et al.* which all solely rely on Bluetooth beacons for proximity confirmation [10, 30, 32]. Levi *et al.* [34] demonstrated a practical attack against these schemes by implementing a Bluetooth relay. Their procedure does not require the attacker to obtain any secret and allows forwarding messages between prover and verifier without alteration. They also found that the relay is not detectable if the implementation follows the Bluetooth protocol definition.

Work on NFC based security tokens indicated that the problem persists independent from the used communication technology. First, Francis *et al.* [13] demonstrated an attack on NFC based contactless transactions using a Bluetooth connection between ghost and leech device. Next, using their setup, Kfir *et al.* [35] established a connection between a smart card and a reader over a distance of 50 meters. Later, Sportiello *et al.* [36] increased this range to a geographical scale without any constraints on the prover and verifier position. Their transmission spanned across different locations over several kilometres using only off-the-shelf hardware.

The relevance of this issue was further emphasised by Francillon *et al.* [11], who demonstrated a working relay attack against a passive key-less entry and start system of a car. The authors investigated two relay methods: relay over-cable and relay over-the-air. While a cable has a bigger range but might raise suspicion, the

wireless system still successfully unlocked the target with 100 meters between key and vehicle. In a scenario, in which the user sits in a coffee shop or office next to her parking spot, this poses a genuine threat.

2.2.2 Spoofing Attack

We identify two types of spoofing attacks: signal-based and data-base spoofing. In both cases, an attacker tries to manipulate a piece of location information to influence the outcome of a location dependent algorithm.

Signal-based Spoofing

Previously mentioned relay attacks are a special case of signal-based spoofing. The signal of a sender is fabricated to convince the target receiver that both entities are within close proximity. This approach works even if authenticated beacons are used as long as the attacker can exploit the legitimate transmitter as an oracle. For unauthenticated signals, such as GPS, it was hypothesised that a signal can be spoofed to lead a receiver to believe that it currently resides at coordinates chosen by an attacker. One of the first works concerned with the feasibility of such an attack was done by Tippenhauer *et al.* [29]. They found that with some limitations on the placement of the adversary's transmitters, the location information of a group of devices can be arbitrarily spoofed while preserving the constellation between them. This is particularly crucial if the devices perform collaborative plausibility checks to defend against targeted spoofing attacks.

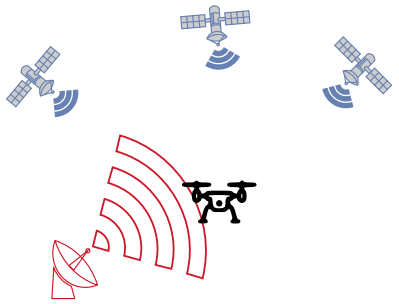


Figure 2.6: Signal-based GPS spoofing attack against a drone in the presence of three satellites. The attacker's transmitter is indicated in red.

The increase in military use of Unmanned Aerial Vehicles (UAVs) combined with customer drones becoming widely available sparked further research in this area. Approaches that can be used to destroy, manipulate, or even take full control of aerial vehicles [37–40] demonstrated the severity of signal-based spoofing attacks.

Their basic methodology is shown in Figure 2.6. Especially due to the availability of modern Software Defined Radios (SDRs) at a low price, it is only a matter of time until these attacks appear outside of a research lab.

Data-based Spoofing

While signal-based spoofing usually requires physical-layer access or custom-made hardware, data-based spoofing can be implemented purely in software and hence has a lower entry burden. For example, the standard for the first generation of connected vehicles mandates every car to broadcast their current location in dedicated awareness messages. A dishonest vehicle could simply provide falsified information to disturb applications like collision avoidance systems [41]. Consequently, any location claim made by a third party has to be treated with appropriate care depending on the trust relationship between sender and receiver.

2.3 Proposed Solutions

As mentioned earlier, for many applications or to perform plausibility checks of received data, confirming the proximity between two devices is sufficient. Proximity can either be determined through a protocol directly or derived from the absolute position of two devices. In this section, we review defence techniques that have been proposed to address aforementioned attacks and provide a secure basis to reason about the constellation of mobile devices. In particular we review distance bounding protocols, schemes that compare the ambient context of two devices, and location confirmation using device trajectories.

2.3.1 Distance Bounding

Distance bounding uses a cryptographic protocol between two devices to establish their physical distance. While this protocol family originates from the work by Brands *et al.* [42] in 1993, a concrete implementation for today's application scenarios was not proposed until 2009 [43–48]. Yet, despite the availability of more modern hardware, the underlying principle remained the same. As shown

in Figure 2.7, a challenge response protocol run between a prover and a verifier forms the basis for the required measurements.

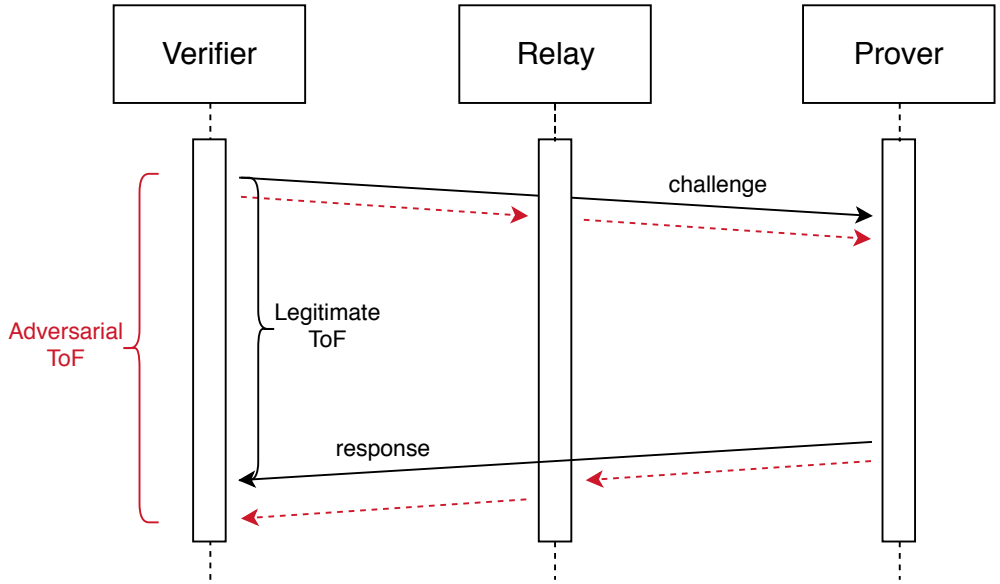


Figure 2.7: Distance bounding protocol under relay attack. The adversary's transmissions are indicated in red, legitimate transmissions in black.

Based on the fact that an electromagnetic wave travels at about the speed of light, the distance between the two parties can be estimated using the Time-Of-Flight (ToF) between the verifier's challenge and the prover's response. A relaying attacker can only increase this delay but never decrease it. Hence, it is possible to appear further away but a prover cannot be *brought closer* than it actually is.

As for the implementation of these protocols, even marginal delays of just one nanosecond already affect the estimated distance by at least 30 cm. Hence, these bursts have to be implemented on the lowest possible communication layer which requires custom-built hardware [44].

For VANETs, Song *et al.* [49] proposed a combination of distance bounding, plausibility checks, and ellipse-based location estimation to verify location claims. While this was evaluated using simulations, a wireless implementation of the protocol is sensitive to interference [44] which poses a major challenge in an environment with a multitude of vehicles. Hence, despite feasible in theory, the hardware requirements make an adoption in smart phones or vehicles unlikely.

2.3.2 Ambient Context Comparison

Another defence mechanism against falsified proximity claims is the use of cues from the surroundings of two devices. The basic idea is simple: if two devices *see* the same environment, they can be considered co-present.

Sustained co-presence was first introduced by Miettinen *et al.* [50] as the continuous comparison of context parameters. In order to secure pairing operations from man-in-the-middle attacks, the authors set out to verify that two devices are within close proximity. They achieved this through periodic comparison of ambient luminosity. However, as experiments showed, indoor luminosity can be influenced by an attacker and external light also varies throughout different seasons of the year.

In their recent work, Han *et al.* [51] aimed to authenticate VANET messages through physical context comparison. The authors proposed to use the road surface as an external factor a sender and receiver should have in common if they are in close proximity. According to their hypotheses, road conditions like bumps and potholes can be measured by an accelerometer and used to derive a shared secret between co-present vehicles. Unfortunately, in their evaluation the entropy of the key material and the effect of different road surfaces remained unexplored making the system's security guarantees unclear.

Halevi *et al.* [52] proposed a co-presence detection based on context comparison using audio and light measurements. Their use case focused on NFC authorised mobile payments which are exposed to relay attacks. To prevent this, the merchant terminal as well as the mobile phone took audio and light probes of their environment and try to match them. They evaluated both modalities separately and achieve a False Accept Rate (FAR) of 6.5% and a 5% False Reject Rate (FRR) for light, while reporting a FAR and FRR of 0% for audio, which makes the result appear untrustworthy.

Similar approaches were explored by Truong *et al.* [8] and Shrestha *et al.* [53]. Both were concerned with ZIA systems and their resilience against relay attacks. The work by Truong uses audio, GPS, WiFi, and Bluetooth signals. They identified WiFi as the dominating context feature with a FAR of 2% and a FRR of 1%. When

facing an adversary capable of manipulating radio frequencies by amplifying or reducing their strength, they recommended the fusion of all monitored data sources. Shrestha, who used temperature, precision gas, humidity, and altitude readings, found that a fusion of the different measurements reduced the initial FRR and FAR from 24% and 33% for an individual modality to 3% and 6%, respectively.

However, in their follow-up work, Shrestha *et al.* [54] demonstrated the manipulation of a wide range of contextual parameters. The result was an increase in FAR to 66% even when the adversary only manipulated a single modality. They hypothesised that approaches which relay on multiple modalities are not *per se* more secure, as it highly depends on the weights assigned to the individual features. These findings emphasise that a thorough analysis of the deployed machine learning algorithms is needed in order to give reliable security guarantees of such systems.

Karapanos *et al.* [55] also investigated context information as a way to improve mobile system security. Instead of preventing relay attacks, they used the audio fingerprint of a location as a second factor for user authentication. The authors implemented a system that used the browser of a computer to record environmental noise and a co-located smart phone to compare the two recordings. The threat model they assumed included a remote attacker who compromised the user's credentials but was not sharing the location of the victim.

2.3.3 Trajectory Recognition

Often, methods to compromise a victim's privacy can be turned around and offer a new way to confirm a device's location. For example, by correlating mobile measurements with location ground truth, such as a street map or the timetable of public transportation, a user's location can be estimated. In the following we review attacks that could be converted to serve as defence mechanisms.

The work by Nawaz and Mascolo [56] explored the significant transportation routes of a user based on gyroscope data. According to their hypothesis, a route exposes a certain signature which can be expressed in angular momentum. They

applied dynamic time warping to account for differences in traffic conditions and were able to re-locate a device on previously trained routes.

Similarly, Narain *et al.* [23] showed that it is possible to infer routes taken by a user, based on permission free on-board sensors, i.e. gyroscope and accelerometer. Using these sensors, they derived bends in the road and turns at intersections to obtain a route fingerprint. Through publicly available map information this sequence was then used to infer the location of the vehicle and subvert user privacy.

Inspired by dead reckoning, Gao *et al.* [24] proposed an approach of using the speed and start location of a vehicle to estimate its final destination and path. By matching driven segments to map data, they achieved an accuracy of 500 meters for a quarter of the tested traces in the New Jersey and Seattle area.

In their work, Watanabe *et al.* [25] identified a user's train trips based on inertial measurements. First, the user's activity is classified into inside a *vehicle*, *walking*, or *stationary*. Afterwards, the transition times between the different modes are correlated with timetables of commuter trains. Each train trip is weighted according to its popularity to reduce the number of candidates. Their results showed that location detection along train networks is feasible.

2.4 Summary

In this chapter we described different ways to acquire the constellation of mobile devices either directly or to use their respective locations in a common frame of reference to confirm their proximity. Next, we presented relay, signal-based, and data-based spoofing attacks. These illustrated the threats faced by mobile location-aware systems when relying on location information. Last, we reviewed current proposals to mitigate these threats and securely determine the proximity of interacting entities. Despite their advantages, these solutions still face problems. Distance bounding requires low-level hardware and is not fit for use in busy wireless environments. Supporters of context comparing systems argue that it preserves user privacy since no absolute location information is shared but just observations of environmental sensors. Yet, as we discussed in Section 2.3.3, some of this

information might still bear the potential of compromising privacy. Lastly, trajectory recognition has so far been considered a threat rather than a solution and hence requires further research efforts.

In the next chapter, we are going to do exactly that. We investigate the use of a device's trajectory as a second factor in a user authentication system. The security of location-aware mobile devices will be the dominating theme for the first half of this dissertation, for the second half, we shift the focus towards their privacy implications.

Be yourself; everyone else is already taken.

— Oscare Wilde

3

Prover-verifier Proximity in Zero-Interaction Authentication Protocols

Contents

3.1 User Authentication 24
3.2 System Proposal 25
3.3 Concept and Assumptions 27
3.3.1 CHAP Integration 27
3.3.2 Application Scenario: Premise Access Control 28
3.3.3 Adversary Model 29
3.3.4 Design Goals 29
3.4 STASH Architecture 30
3.4.1 Trajectory Representation 30
3.4.2 Primitive Generation 32
3.4.3 Trajectory Comparison 34
3.4.4 Comparison Thresholds 36
3.5 Performance Evaluation 38
3.5.1 STASH Prototype 39
3.5.2 Movement Recognition 41
3.5.3 Experimental Data Acquisition 41
3.5.4 Path Separability 41
3.5.5 Determining Initial Thresholds 43
3.5.6 Impact of Reference Path Length 45
3.5.7 Multi Path Learning 46
3.5.8 Energy Consumption 48
3.6 Security Analysis 51
3.7 Discussion 53
3.8 Summary 54

After having discussed the different threats faced by location-aware mobile systems and current solutions, we set out to investigate the role of location information in user authentication. In this chapter, we first describe a new generation of authentication protocols that rely on a user's proximity when confirming her identity. We identify their vulnerability against relay attacks which prevents the secure adoption of these schemes in the real world. Then we present our solution to mitigate these attacks by providing a complementary component that can be easily integrated into existing systems. To validate the feasibility of our proposal, we implement a prototype and conduct experiments showing its usability and security. This chapter is based on [57] and [58].

3.1 User Authentication

Protecting data or physical objects from unauthorised access requires authentication the process of confirming a user's identity. For this, even today, the predominant principle remains the exchange of a secret in conjunction with an identifier specifying the claimed identity.

The two parties that participate in the authentication protocol are the *prover*, the entity that claims the identity, and the *verifier*, the entity that decides on the validity of the claim. Before the verifier is able to check the credentials provided by a prover, both run through an enrolment procedure. During this process, the secret is associated with the user specific identifier and made available to both parties. Secret and identifier make up the credentials of the user.

Throughout recent years, credentials of different manifestations have been proposed and are being implemented in real-world applications: passwords [59], biometric traits, and user behaviour [60], as well as token based credentials [31]. The latter are commonly deployed as a second factor in two-factor authentication protocols [61], while passwords and biometrics are usually used as a primary factor. Even though biometrics have been in the focus of research for decades, password

based systems [59] still remain the prevalent method for user authentication. This is particularly intriguing given their known disadvantages such as having to memorise a sequence of characters, leading to users writing them down, and requiring manual input, causing users to choose simple phrases.

In a pursuit of addressing these issues, Zero-Interaction Authentication (ZIA) [9] emerged as an approach to obtain credentials seamlessly from the user. Its aim is to strengthen authentication through an improved user acceptance by making user interaction obsolete and replacing passwords with a security token. One way to implement ZIA is by using the availability of a wireless connection between prover and verifier as an indicator for the user's intent to authenticate. Once the authentication procedure is initiated, credentials are exchanged, for example between a smart phone and a computer using Bluetooth, or between a smart tag and a door lock using NFC. The result is a transparent and effortless approach which replaces human-to-machine interaction with machine-to-machine communication.

Although such a scheme satisfies the requirement for zero-interaction, it also introduces the wireless channel as new attack surface. As presented in the previous chapter, an attacker who relays messages can subvert the implicit assumption of a user's proximity based on wireless connectivity. By bridging the transmission gap between the prover and verifier, the range of the wireless connection is extended and unintended authentication attempts can be initiated [11, 13].

3.2 System Proposal

We present STASH, an approach for thwarting relay attacks on wireless ZIA schemes: the prover transmits authentication credentials only if it can confirm its proximity to the verifier. The resulting system provides relay-resilient transparent authentication. For proximity verification, the prover compares its approach trajectory towards the intended verifier to a reference trajectory previously authorised by the user. In order to be independent from external signal sources, these trajectories are captured using low-cost sensors commonly available on personal devices, e.g. Micro-ElectroMechanical System (MEMS) on smart phones.

Monitoring the user's movement purely through on-board sensors of the prover device protects the verification process from adversaries who can influence environmental factors. As we recall from Chapter 2, GPS and other radio signals can be injected or modulated while ambient properties such as sound, temperature, or lighting can be influenced through devices that introduce noise or absorb parts of these signals. Additionally, individuals who rely on STASH do not risk exposing their whereabouts since their trajectories are not shared with a third party. By analysing empirical data collected using a STASH prototype, we demonstrate the security of STASH against relaying adversaries and investigate its ease-of-use. STASH is energy efficient and can be easily integrated to complement existing ZIA schemes by only requiring modifications on the prover. The contributions made in this chapter can be summarised as follows:

- We propose using prover-side proximity verification to resist relay attacks against zero-interaction authentication schemes that use device proximity as an indicator of a user's intent to authenticate.

- We design and implement STASH, a system incorporating this idea by addressing several challenges when capturing a prover's approach trajectory and comparing it to previously authorised trajectories.

- We conducted experiments using an Android implementation of STASH and gathered trajectory data of 20 different routes in two cities (totalling 123 km). Using this dataset, we evaluated FRRs and FARs of the system to demonstrate the security and usability of STASH.

- We show that STASH's average energy consumption is low: we estimate that under typical usage conditions, the battery drain due to STASH over the course of one work day is in the range of 4%-7% of the total capacity for a modern smart phone battery.

3.3 Concept and Assumptions

For a better understanding of how STASH can be integrated into existing ZIA systems, we first review an abstracted version of their architecture. We highlight how STASH complements the protocol flow to provide prover-side proximity verification. For our analysis, we focus on a particular application scenario which is presented before we outline the capabilities of the attacker we consider. Lastly, we define a set of functional and non-functional requirements that we will refer to towards the end of this chapter.

3.3.1 CHAP Integration

The underlying principle is a standard Challenge-Handshake Authentication Protocol (CHAP) during which a personal device owned by the user receives a challenge from a verifier devices. Using a *previously established security association*, e.g. a symmetric key shared between the two devices, the personal device generates a response and sends it back to the verifier. After checking that response's validity, the verifier signals the outcome of the protocol run. If that response is valid, the verifier considers the personal device authenticated.

For proximity-based transparent authentication, two components are additional to this design. Figure 3.1 gives an overview of the resulting architecture where steps 2, 4, 5, and 7 resemble the challenge-response protocol. The first additional component is used in step 1. It enables the verifier V to sense the proximity of P, for example by checking for the presence of P's MAC address in the local Bluetooth environment [10]. In step 5, V uses the second added component to determine its proximity to the verifier. For example by measuring the RSSI of received packets or their ToF [32, 44]. However, by deploying a ghost and leech device, this process is vulnerable to relay attacks as V would measure the packets it exchanges with the leech instead of the actual personal device of the user.

To protect against such attacks, we improve this design by having the prover guard access to the authentication credentials by first verifying its proximity to the verifier. We propose that the prover performs this proximity verification by

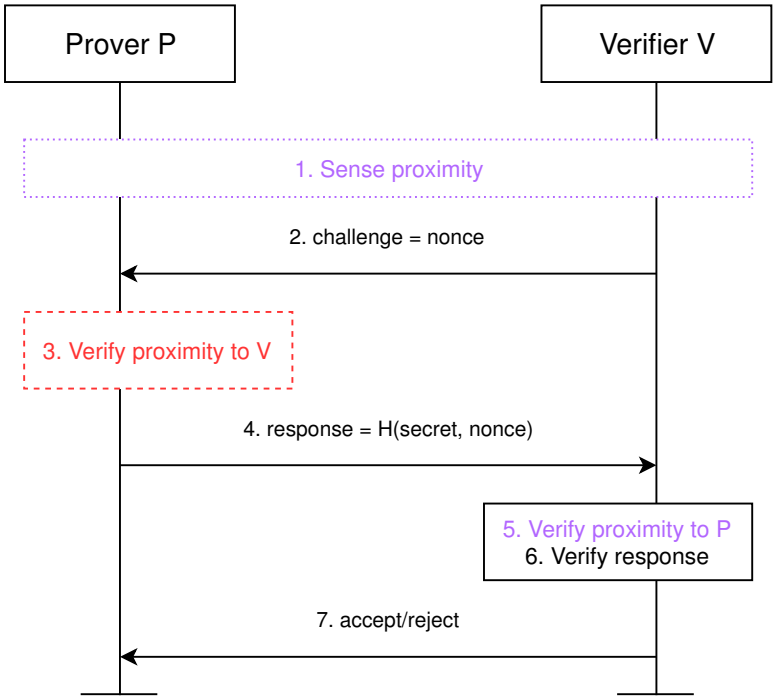


Figure 3.1: Architecture of a wireless ZIA scheme between a prover P and verifier V. The CHAP is shown in black while the components required to transform it to a zero-interaction protocol are indicated in purple. The proximity verification component introduced in our approach is shown in red.

examining its approach trajectory towards the verifier. In several ZIA application scenarios, this is an *inherent repetitive pattern* since the user first has to travel to the location where authentication is required. We assume that the communication between the two parties is secured using for example Transport Layer Security (TLS).

3.3.2 Application Scenario: Premise Access Control

We envision a scenario where a person accompanied by a prover device, e.g. a smart phone or vehicle, routinely approaches an access controlling barrier. The barrier, acting as the verifier, will open to let the user pass after transparently authenticating her personal device. As a fallback mechanism, if proximity verification fails, the user may still open the barrier through explicit proximity confirmation on the prover. For example by unlocking her smart phone and approving the authentication request. One instance of such a scenario is a person in a vehicle such as a car, bicycle, or wheelchair requiring easy and fast access through a gate

or door at her home or workplace.

As another example, traffic lights could offer a faster red-green cycle depending on the presence of vehicle at adjacent roads to increase efficiency. Similar to a pedestrian crossing, the vehicle would trigger a routine to indicate its presence. By requiring the vehicle to authenticate with the traffic light, Denial of Service (DoS) attacks could be mitigated as misbehaving vehicles could be denied access.

3.3.3 Adversary Model

We consider an adversary, who has deployed a wireless relay providing him with Dolev-Yao [33] capabilities. By relaying the challenge and response and thus artificially extending the range of the wireless channel, the attacker can successfully bypass the verifier's proximity verification. Regardless whether the verifier uses RSSI or ToF, it will be measuring its interaction with the ghost device rather than that of the responding prover. As discussed in the previous chapter, such attacks are feasible in practise as was demonstrated for several scenarios [11, 13].

We do not consider adversaries who gain physical access to the prover device, e.g. through theft. Continuous user authentication techniques, such as biometric authentication [62] can ensure that it is the legitimate user who is in possession of the device. Hence, we implicitly identify the location of the prover device with the one of the user throughout the remainder of this chapter. We also assume that the attacker has not infected the prover device with malware or any other means that give him access to the pre-shared secret.

3.3.4 Design Goals

We now briefly define the goals for our relay-resilient proximity verification system. In our analysis later, we are going to reference these to indicate their fulfilment.

R1. **No external signals**: Since an attacker could manipulate ambient properties, proximity verification must not depend on any external signal.

R2. **Local decision-making**: The prover's proximity verification must be carried out entirely on the personal device. There are two reasons for this requirement. Firstly privacy, data collected for proximity verification must not be exposed to any third party as it may be used to infer the location history of the user. Secondly deployability, a local solution can be integrated more easily into existing ZIA schemes by only modifying provers without affecting the protocol or the implementation of verifiers.

R3. **Security**: The system must not incorrectly conclude that the prover is near the verifier particularly in the presence of a wireless relay. Therefore it must minimise the False Accept Rate (FAR).

R4. **Usability**: The goal of transparent authentication is to make user action obsolete. Yet, if trajectory comparison fails when the user is in fact near the verifier, she will be required to fall back to explicit proximity confirmation. Our system must therefore minimise the False Reject Rate (FRR).

R5. **Efficiency**: The computational and therefore energy cost of proximity verification must be small to not diminish the user experience.

3.4 STASH Architecture

Designing STASH to work on commodity devices raised several challenges leading to questions like "how to effectively represent a trajectory using only accelerometer/gyroscope measurements?" and "how to best compare these trajectory representations?". From these, we identified the following four tasks for our system: representing trajectories, capturing trajectories, comparing trajectories, and determining comparison thresholds.

3.4.1 Trajectory Representation

To describe someone's trajectory on Earth, we could represent a person's movement as a sequence of coordinates, e.g. within the geographic coordinate system. However, to satisfy requirement R1, unauthenticated data sources, such as GPS [29] cannot

be used. Dead reckoning offers a solution which only relies on gyroscopes and accelerometers to compute a sequence of coordinates [28]. Unfortunately, this approach is subject to cumulative errors and a missed turn can change the resulting sequence completely [63].

In order to be more resilient against these types of errors, we represent a trajectory as a *temporally ordered sequence* of primitives consisting of segments of movement interleaved with left and right turns. Figure 3.2 shows an example of a prover P approaching the verifier V and how this trajectory can be partitioned into discrete components.

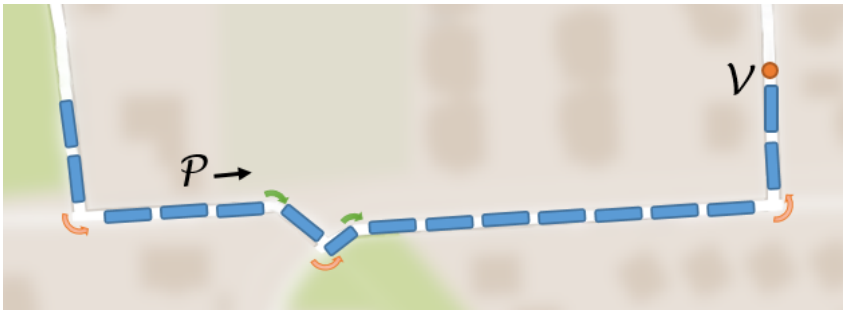


Figure 3.2: P's approach trajectory can be described by a set of primitives needed to reach V. In this example, the primitives are: *move* 2, *left* 90°, *move* 3, *right* 30°, *move* 1, *left* 90°, *move* 1, *right* 30°, *move* 7, *left* 90°, *move* 2.

Using sensor data, we recognise two streams of primitives: $(M/S, t_i)$ for "movement" or "stationary" at time t_i, generated at a fixed rate and $(L/R, t_i)$ for "left turn" or "right turn" at time t_i, generated opportunistically whenever a turn occurs. These two streams are merged into one sequence, with turn events taking precedence. For the trajectory in the example presented in Figure 3.2, a sequence as shown in Equation 3.1 would be produced. Exponents indicate sequential repetitions of the same symbol.

$$\begin{aligned} Seq = {} & (M^2, t_0), (L^9, t_1), (M^3, t_2), (R^3, t_3), (M, t_4), (L^9, t_5), \\ & (M, t_6), (R^3, t_7), (M^7, t_8), (L^9, t_9), (M^2, t_{10}) \end{aligned} \tag{3.1}$$

3.4.2 Primitive Generation

When deriving movement primitives from low-cost MEMS sensors, it has to be taken into account that these are noisy and are temporarily turned off for power optimisation. According to previous studies, these sensor errors mostly occur due to (device-specific) constant biases and high-frequency white noise [64]. As a result, time series obtained from them are often unreliable and contain temporal discontinuity.

In the following, we describe how we overcome these challenges to detect *movement*, *stationary*, *left turn*, and *right turn* events. The data flow for the generation algorithm is shown in Figure 3.3. Once identified, events are associated with a timestamp which enables the composition of the final trajectory as a temporally ordered sequence.

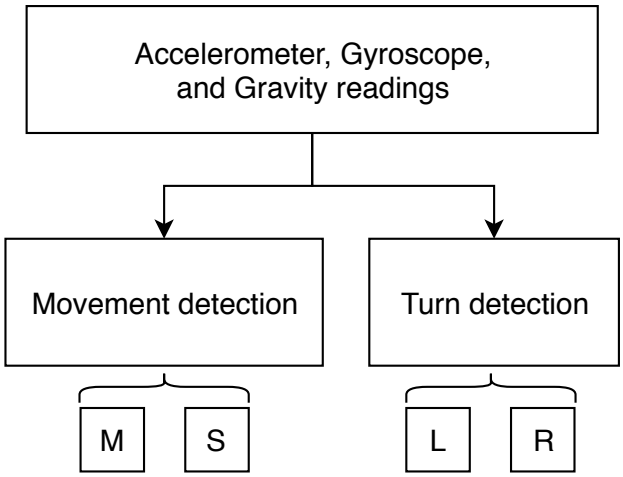


Figure 3.3: The data flow of the movement capturing system. The two Inertial MEMS sensors and a third software sensor (gravity) provide the data for the movement and turn detection algorithms. These produce movement M, stationary S, and turns L, R primitives.

Turn Primitives

Turn detection relies on two data sources. First, gyroscope data which provides angular speed (rad/s) in all three dimensions when the prover device is rotated. Second, a three dimensional vector which indicates the device's orientation offset with respect to the Earth's gravity field represented by a vector pointing perpendicular towards the ground plane.

We begin with a pre-processing step, during which drift is removed using a high-pass filter, where gyroscope measurements smaller than 8.6° are exponentially weighted down. Next, we use the gravity offset vector to correct for the orientation of the measuring device and project the three dimensional (x,y,z) gyroscope time series such that the z-component is pointing opposite the gravity vector. Finally, the heading angle is obtained by integrating the angular speed around the z-axis. Using the resulting time series, turns are recorded when the two seconds sliding window of that data's standard deviation is above a threshold ($\sigma_1 = 3°$). To achieve a finer detection of the start and end of a turn, a sliding window over the standard deviation with a smaller threshold ($\sigma_2 = 1°$) is applied once a turn has been detected.

One assumption of the turn detection system is that the recording device provides reliable gravity estimates. To ensure this, the device could, for instance, be integrated into a vehicle or firmly attached to the body of the user to avoid perturbations. For smart phones we noticed that orientation changes cause errors in turn estimation[1]. The gyroscope data is unreliable in such situations and hence discarded. During the experiments for the evaluation of our system, we controlled for this problem by integrating the prover device into a bicycle.

Finally, to increase the resilience against sensor noise, we discretise turns using their angle values, i.e. a fixed number of degrees is equivalent to one symbol. After exploring different sampling rates and sensors, we found that a 20 Hz sampling rate can consistently detect turns with a precision of 15°.

Movement Primitives

For movement detection, we use Logistic Regression (LR) [65] that is applied to the three dimensional accelerometer readings of the prover device. While this algorithm can predict the movement state of the device in real-time, it is memoryless, thus not taking previous results into account. In reality however, two successive events may depend on each other. For example once a person started walking, it is more likely that she continues walking after the first few seconds rather than stopping abruptly.

[1]On the Android platform, gravity is a software sensor (a low-pass filter on raw accelerometer data) which takes a few seconds after an orientation change to stabilise.

Similarly, once a person sat down, she usually sits down for more than just a few seconds. We deploy a Hidden Markov Model (HMM) to capture this dependency.

In HMMs, probabilities of transitions between hidden states are modelled with a first-order Markov chain. Each hidden state gives a cue about itself by emitting a primitive at any given time. For our application, we want to determine movement primitives (M or S) by observing the output of the LR. We use a HMM Viterbi algorithm [66] to smoothen the observations into the most likely sequence of primitives. For each group of five consecutive measurements, a representative is chosen based on the dominating state, for example if four M and one S are observed, the S symbol is overruled. This method gives us a continuous stream of a M or S primitive for five second intervals.

3.4.3 Trajectory Comparison

For the following we define two types of trajectories. *Reference paths* are previously authorised trajectories taken by the user when approaching a verifier while carrying the prover device. *Candidate paths* are trajectories that have been recorded by the prover just before an authentication attempt. To verify its proximity to the challenging verifier, the prover compares a candidate path to the reference paths associated with that verifier. This comparison can be carried out entirely on the prover device which satisfies requirement R2.

Figure 3.4 shows how the trajectory comparison enables the proposed proximity verification and integrates with the ZIA protocol. If a trajectory comparison is successful, the prover continues with the authentication protocol by computing the response to the challenge and sending it to the verifier. Otherwise, the system notifies the user by offering explicit proximity confirmation. In case the user manually approves the authentication run, STASH can add the current candidate path to its repository as an authorised trajectory. Multiple authorised trajectories for the same verifier form a cluster, so we choose the best representative among those paths for future comparisons. Through pairwise comparing all stored trajectories, the instance with the least distance to all other paths, i.e. the medoid, is selected.

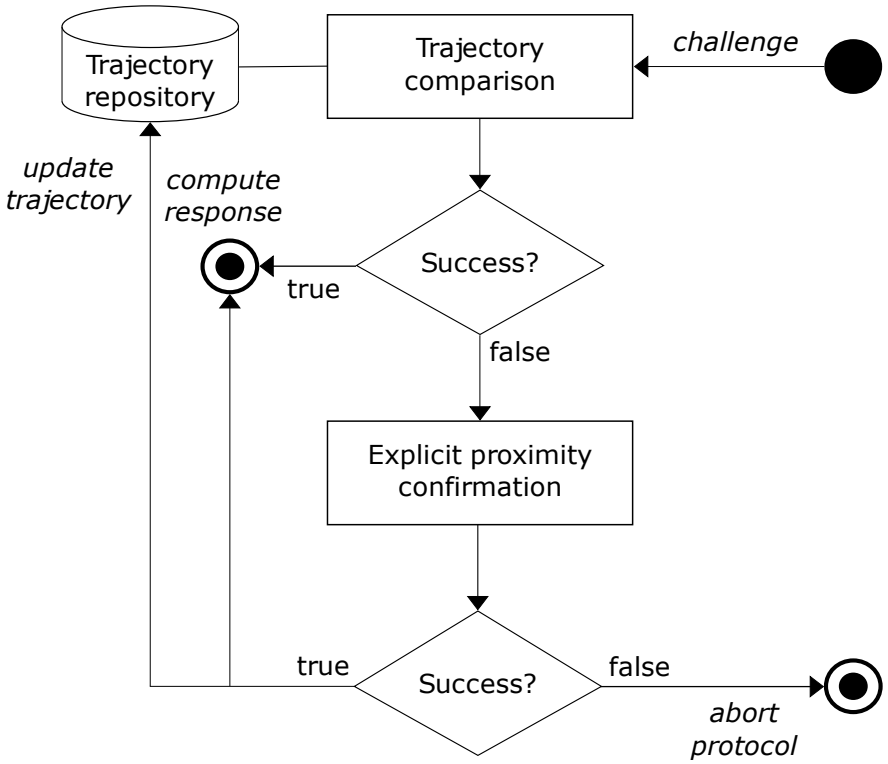


Figure 3.4: Prover-side proximity verification: The prover compares the current candidate path against reference paths in the repository. It only computes the response to the verifier's challenge if the paths match. Otherwise the protocol is aborted unless the user explicitly approves the authentication run.

As for trajectory comparison, candidate and reference paths are represented as sequences of characters. Therefore, this task is equivalent to determining the similarity between strings, which is a well established area of research.

One common metric for string comparison is the Levenshtein distance. It represents the minimum number of *inserts*, *replace*, and *delete* operations required to transform one string into the other. A more advanced algorithm, the Damerau-Levenshtein distance additionally allows *transposition* of adjacent characters. Both algorithms are commonly used in spellcheckers [67].

In bioinformatics, a different approach was chosen for the algorithms of Needleman-Wunsch (NW) and Smith-Waterman, which are used to align DNA sequences either globally or locally [66]. The scoring schemes of these algorithms add a reward for correct symbols, while *inserting*, *replacing*, or *deleting* a symbol inflicts a penalty. We chose to investigate the feasibility of these alignment algorithms for our application

for two reasons. Firstly, similar to our trajectory representation, DNA sequences are based on a small set of characters. Secondly, since pieces of DNA can be made up of thousands of elements, these algorithms are designed to operate on long sequences.

After a preliminary evaluation, we chose to use the NW similarity[2]. In NW terminology, both insertions and deletions are called *gaps*. We chose the parametrisation: match *+1*, mismatch *-2*, and gap *-1*. Before each comparison, S symbols are removed and timestamps are used to trim candidate paths to the same temporal length of the reference paths stored for the challenging verifier.

Due to noise from various sources, as mentioned earlier, legitimate candidate paths of the same reference path may differ slightly. Hence, we need to establish a decision threshold that determines how much noise is acceptable. If the similarity score is higher than the threshold, the candidate path is accepted, otherwise it is rejected. This introduces a trade-off between usability (FRR) and security (FAR). In Section 3.5.4 we show how experimental data can be used to determine an initial threshold that provides a balance between the two. However, depending on the application scenario, different emphasis can be put on R3 and R4 by shifting the threshold towards a lower FAR or a lower FRR, respectively.

3.4.4 Comparison Thresholds

As mentioned in the previous section, a decision threshold is required to compensate for variations in a trajectory's representation. In order to compute an adequate value for this parameter, we rely on two types of thresholds: an *initial* and a *local* threshold. While the latter aims to improve the performance of the trajectory comparison, i.e. lowering the FRR without increasing the FAR, the former is intended to simplify the enrolment procedure for the user.

After a user enrols a new verifier, STASH's trajectory repository contains only one instance of the path leading towards that verifier. In order to still be able to perform proximity verification, the initial threshold is used to determine whether or not a candidate path is similar to the reference path.

[2]Needleman-Wunsch had the best FAR/FRR trade-offs among the tested algorithms on our dataset in Section 3.5.

Yet, this initial threshold might under- or overestimate the variation in future instances of a reference path and hence be too strict or permissive, respectively. Therefore, once STASH is in operation and several protocol runs have been handled by the system, it can employ user feedback to continuously improve its performance. In particular, if trajectory comparison fails, the system gains information after the user explicitly confirms the authentication attempt. To incorporate this knowledge and improve future decision making, we propose the path-specific local threshold.

The computation of a local threshold for a reference path R leading to a verifier V requires instances i_R of R and instances $i_{\bar{R}}$ of paths $\bar{R}$ that do not lead towards V. These are used to calculate *within-* and *between*-class similarities based on which a threshold value is selected. Yet, when a user just started using STASH we do not have any instances i_R, $i_{\bar{R}}$ to compute these similarities. While instances i_R will be accumulated as the user repeatedly traverses R, the requirement for $i_{\bar{R}}$ is to cover a larger number of a user's unauthorised trajectories that resemble her daily movement habits. In order to obtain instances $i_{\bar{R}}$, we create a generative Markov chain from historic movement patterns.

Since the size of the sample set used to compute the local threshold is initially small, immediately switching from the initial threshold to using the local threshold would make STASH susceptible to over-fitting and ultimately degrade the verification performance. Therefore, we propose to initially only assign a small weight to the local threshold and a large weight to the initial threshold. Once more instances of the reference path were observed, the system gradually shifts this ratio and puts more emphasis on the local threshold. We model this *mixed* threshold value d_{mix} as a mixture model using a convex combination [65] of the thresholds d_i (initial) and d_l (local) with a confidence factor $\lambda \in [0, 1]$:

$$d_{mix} = \lambda \cdot d_l + (1 - \lambda) \cdot d_i \tag{3.2}$$

For $\lambda = 0$ we have no trust in the calculated path-specific threshold, and similarly if $\lambda = 1$ we have reached a state with full confidence in the path-specific threshold. In theory this should occur when the system has observed all possible valid path

variations of a reference path, and we believe no additional individual observation will change the decision threshold. A common way to model the confidence in small sample sizes is to use add-one smoothing [65]. Let n be the number of observed instances of a reference path, we then model λ as:

$$\lambda(n) = \frac{n-1}{n} \tag{3.3}$$

This fulfils our boundary conditions for λ with $\lambda(1) = 0$, i.e. we solely rely on the initial threshold, and $\lambda(\infty) = 1$, i.e. we fully rely on the local threshold. Figure 3.5 shows how the confidence factor increases with respect to the number of observed reference path instances. We are going to evaluate the three thresholds mixed, local, and initial in Section 3.5.7.

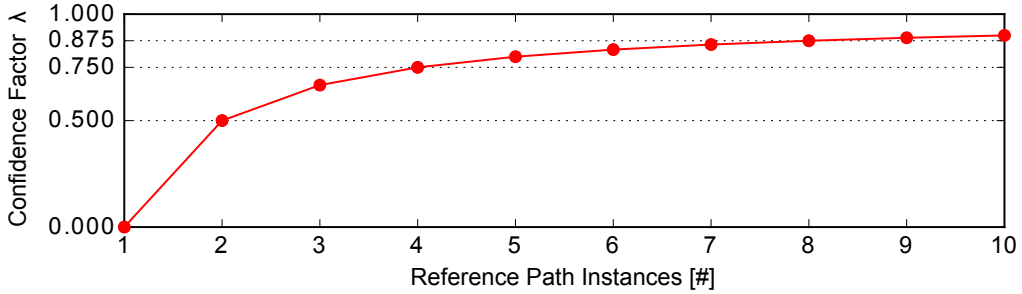


Figure 3.5: The confidence factor λ increases with the number of observed instances of the same reference path. For every two new observations, its distance to one is cut in half, i.e. the confidence grows logarithmically.

3.5 Performance Evaluation

In this section, we evaluate the performance and resource requirements of STASH. First, we present a prototype implementation on the Android platform. Using this application, we collected preliminary data to confirm the parameters of the movement recognition algorithm before acquiring data in a larger field study. Based on the real-world dataset, we show how initial, local, and mixed similarity thresholds can be derived. Next, we present how the length of a reference path and multiple instances of the same path affect STASH's performance. We conclude with an analysis of the energy consumption of the developed application.

3.5.1 STASH Prototype

Figure 3.6 provides two screenshots of the STASH prototype. The screen shown in Figure 3.6a displays outputs of the real-time classification algorithm which determines the movement state of the device in one second intervals. A button at the bottom of the screen allows the user to lock the verifier in case a false accept occurs. Figure 3.6b depicts prompt presented to the user after a failed trajectory comparison. Through manual confirmation, the current trajectory is added to the repository and the pending authentication attempt is approved.

Our STASH implementation requires Android API level 19 or higher and utilises several support libraries, most notably Weka [68] and a Java HMM implementation[3] to realise the classifiers[4]. Other libraries in use are the Lightweight-Stream-API[5], EventBus[6], and ZXing[7].

Storage Requirements

STASH expects a continuous stream of raw accelerometer and gyroscope data to extract the current trajectory. It re-samples the input data to 20 Hz before storing it into a memory buffer (2.3 MB/h each with 32 bit precision), which is a fixed circular buffer holding the past hour of measurements. The values are classified on-demand with LR and the classification results are stored into a separate circular buffer (3.6 kB/h with 8 bit integers). By directly calculating gravity and linear acceleration values from raw accelerometer data, STASH's memory buffer requirements could be reduced to less than 5 MB in total.

Processing Requirements

During data collection, STASH acquires a wakelock to ensure that the power optimisation deployed by modern smart phones does not diminish the quality of the data provided by the phone's MEMS sensors [69]. However to prevent unnecessary

[3] https://code.google.com/archive/p/jahmm
[4] https://github.com/rjmarsan/Weka-for-Android
[5] https://github.com/aNNiMON/Lightweight-Stream-API
[6] https://github.com/greenrobot/EventBus
[7] https://github.com/zxing/zxing

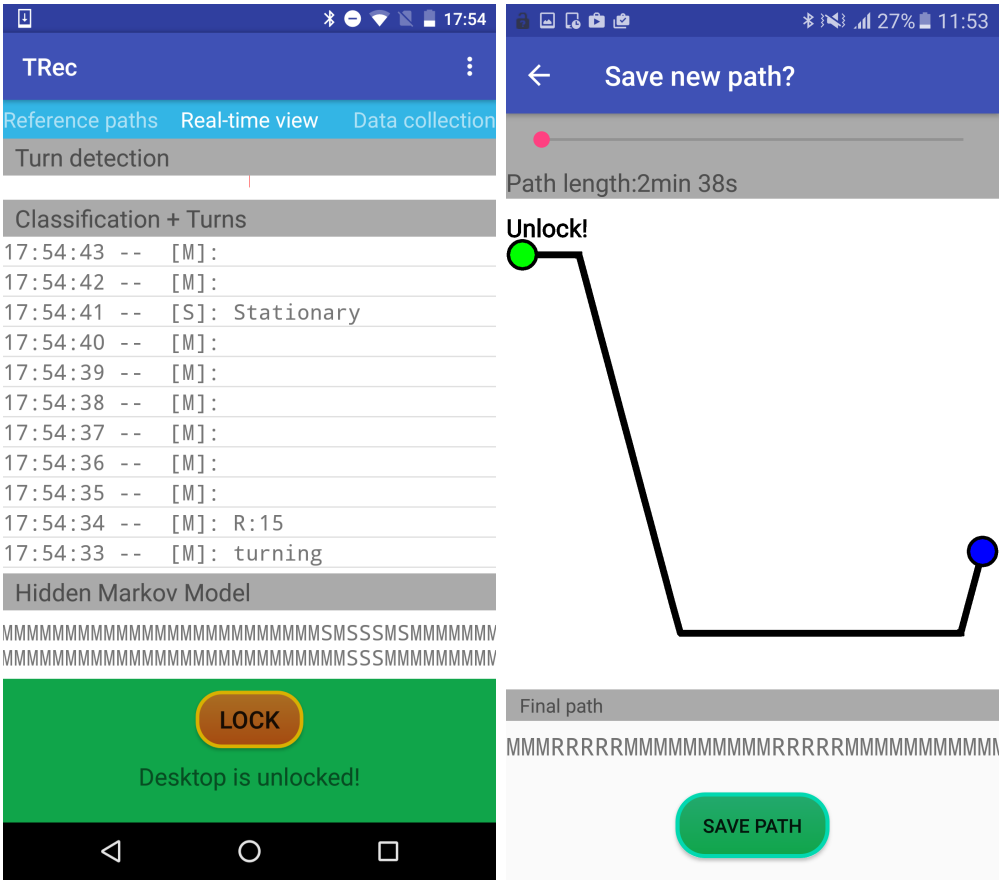


Figure 3.6: Android prototype of STASH: (a) Movement is classified in one second intervals. A button allows the user to lock the verifier in case of a false accept. (b) Interface to approve authentication attempts of unrecognised paths. The temporal length of the reference path can be adjusted using a slider at the top of the screen. The blue and green circle indicate the start and finish of the current trajectory.

battery drain, the application releases the wakelock after the device was stationary for at least 5 minutes to allows normal sleep cycles. Further, data is only pre-processed when trajectory comparison is needed. Once the authentication protocol is initiated, the trajectory is continuously compared at one second intervals for a total of 10 attempts. If successful, the trajectory data is stored and a response is generated. Otherwise the authentication attempt is deferred until the user explicitly confirms the device's proximity to the verifier or a timeout of two minutes causes an abort of the protocol.

3.5.2 Movement Recognition

Based on a preliminary dataset covering different motion conditions over four hours, we confirm the parameters and evaluate the accuracy of the movement recognition algorithm. First, we applied noise-based regularisation to reduce potential over-fitting [65]. We then evaluated the resulting LR model using 5-fold stratified cross-validation to separate training and testing data.

Out of 10 features based on the three dimensional accelerometer and gyroscope readings, we found that the three most significant features were: the standard deviation of the 5 second and 1 second sliding window of the differentiated accelerometer values, and the peak-to-peak amplitude for the 5 second sliding window of the gyroscope data. Evaluation results for LR showed a true positive rate of 98% for movement and 92% for stationary.

3.5.3 Experimental Data Acquisition

We collected real-world data by repeatedly traversing a series of routes to evaluate the accuracy of STASH as a whole. The dataset consists of paths corresponding to 20 different 6 to 12 minute long routes in Espoo, Finland and Oxford, United Kingdom. In order to account for differences in measurement devices, we gathered data using five smart phone models[8] at a sampling rate of 200 Hz. We attached the measurement devices to bicycles to avoid uncontrolled orientation changes. Each trajectory was repeated between 7 and 11 times. Routes contain real-world obstacles, such as traffic lights and crowds, and were conducted on different surfaces, such as gravel, asphalt, and cobblestone roads. In total, the 7.7 GB dataset makes up for 38.6 hours of recording, collected over a distance of 123 km.

3.5.4 Path Separability

For proximity verification to work, it is necessary that the NW algorithm is capable of separating candidate paths into instances of a stored reference path and those that are not.

[8]OnePlus One, Nexus 6, Moto G (gen 2), Nexus 5X, and Samsung GS6.

We use the 20 routes $R_k, k \in \{1, \ldots, 20\}$ of our dataset for which we collected between 7 and 11 instances i_{R_k}. In order to evaluate the within-class similarities of each route and determine if the NW measure provides consistent classification, we calculated the similarities between all pairs of instances, trimmed to a duration of 2 minutes[9]. For each route, this results in at least 21 unique pairwise within-class and 1,595 unique pairwise between-class similarities.

Figure 3.7 shows one instance of within-class and between-class similarities for one route in our dataset. All other routes exposed a similar behaviour. In an ideal noiseless case, all instances i_R of a route R are identical, but under real-world conditions, sensor noise results in a spread of similarity scores. While there is an overlap of within-class and between-class similarities scores in the range $[18, 21]$, most cases can be separated. This makes $18, 19, 20$, and 21 candidates as threshold value. Further, the separability confirms that the NW algorithm provides an adequate similarity measure.

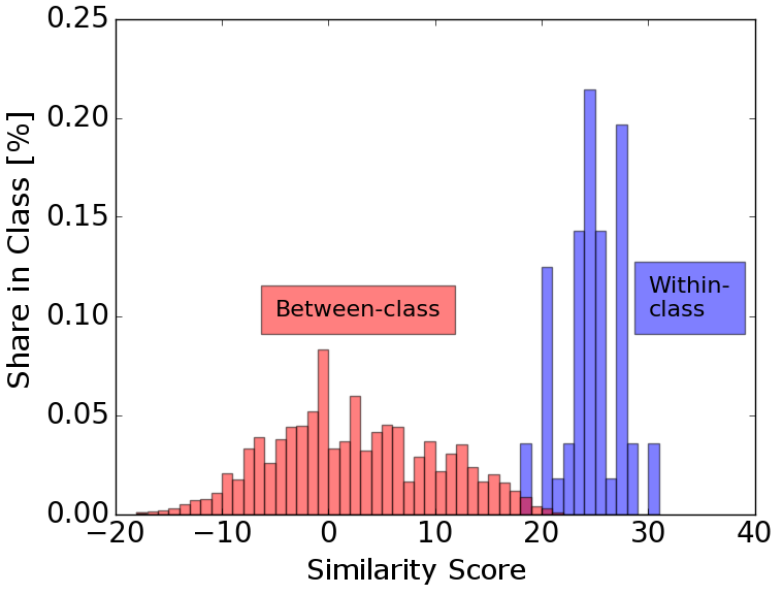


Figure 3.7: An example of within-class (blue) and between-class similarities (red) for one reference path. Choosing a reference path-specific threshold of 18 minimises EER on this route.

[9]We chose this specific path length since it lead to the best Equal Error Rate (EER) as will be presented in the next section.

3.5.5 Determining Initial Thresholds

As described in Section 3.4, the decision threshold that STASH uses can be adjusted based on the number of instances observed for a reference path. However, if only a single instance is known to the system, STASH uses the *initial threshold*; as more instances are collected, their similarities to the reference path are used to compute a *local* threshold, and subsequently the *mixed* threshold. This combination of the two thresholds relies increasingly less on the initial threshold but more on the local threshold as the number of reference path instances increases.

We can compute threshold candidates based on a specific value of the FAR and FRR trade-off parameter α. In order to select the actual value, we choose the threshold exposing the minimal combined error rate err_{cb} defined in Equation 3.4.

$$err_{cb} = \alpha \cdot \mathrm{FRR} + (1 - \alpha) \cdot \mathrm{FAR} \tag{3.4}$$

Figure 3.8 shows optimal achievable FARs and FRRs for pooled[10] reference paths. For each value of α, the minimum combined FAR/FRR is found by varying the decision threshold. In scenarios where the usability of the system is more important, larger values for α should be chosen, while for security critical systems, the opposite is recommended.

Reference Path Length

Given that trajectory deviations caused by systematic sensor noise can occur at any point in time, the number of inaccuracies a valid candidate path may exhibit is correlated to the length of a trajectory. Hence, we suggest to use a variable initial threshold function parametrised with the temporal length L of the considered reference path. We call this function the length dependent threshold d_L which minimises the combined error rate (Equation 3.4) for $\alpha \in \{0.1, 0.2, \ldots, 0.9\}$. After computing these for $L \in \{1, 2, \ldots, 6\}$ minutes, we obtain 6 dependent-independent variable pairs per α, i.e. 9 linear regressions [65]. We

[10] All reference-path specific within-class distances are aggregated to one group, and all reference-path specific between-class distances to another group.

found that the relationship between optimal decision threshold and the length of the path is affine. Hence, after combining the regressions for $\alpha^* = 0.5$ – as indicated in Figure 3.8 $\alpha^* = 0.5$ is closest to the EER point – we obtain the length specific decision thresholds as shown in Equation 3.5. When implementing the STASH prototype, we used these decision thresholds rounded to the closest integer, e.g., $D^*(1) = 8$, $D^*(2) = 18$, and $D^*(5) = 47$.

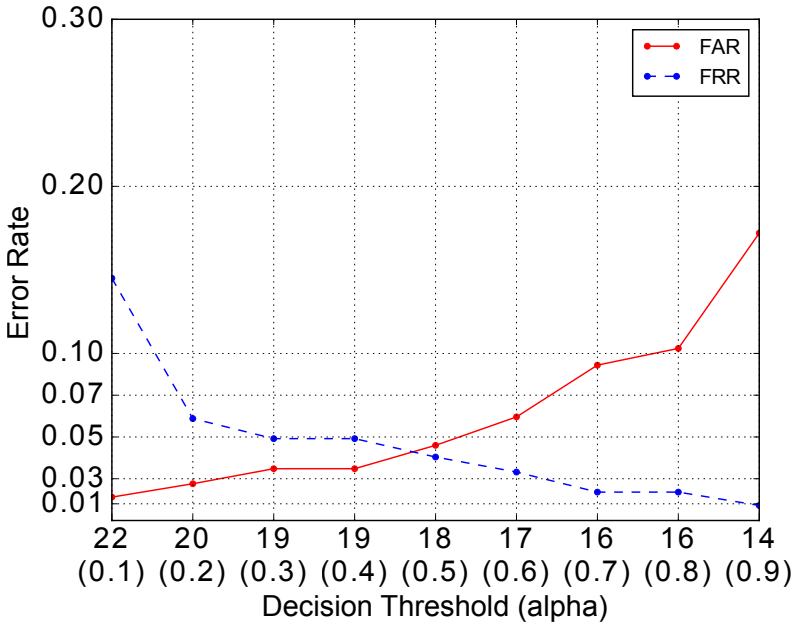


Figure 3.8: False accept and false reject error rates depending on the value chosen for α. For each α, the initial threshold that yields the lowest combined error is given. Large α values result in low FRR, while small values result in low FAR.

In order to evaluate the performance of this length specific initial threshold, we calculate the respective value for each of the 20 routes in our dataset. Figure 3.9 shows the resulting FARs and FRRs (there are 20 lines in total). As can be seen, an EER below 5% can be achieved with a decision threshold set to 18.

$$D^*(L) = 9.69 \cdot L - 1.40 \tag{3.5}$$

In the next two sections, actual test error rates for specific paths that use individual, mixed thresholds are provided. These are evaluated with respect to two different parameters: the length and the number of observed instances of a reference path.

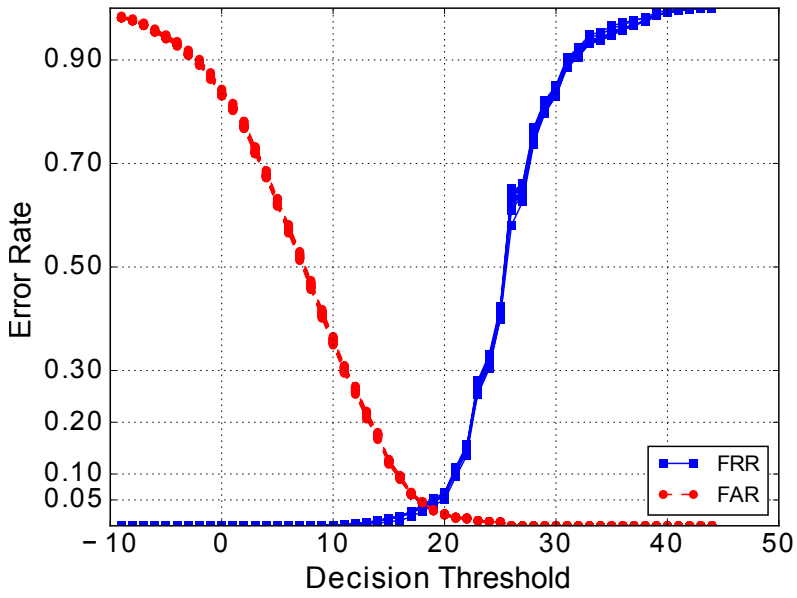


Figure 3.9: Based on the length specific initial threshold values, we compute the error rates FAR (red) and FRR (blue) for 20 routes (20 lines). The testing set for each route consists of paths of all other routes.

3.5.6 Impact of Reference Path Length

The mean error rates and standard deviations for our 20 routes using the length dependent initial threshold are shown in Table 3.1. To evaluate the impact of the reference path length, we compute these values for $L \in \{1, 2, \ldots, 6\}$. FRRs are calculated pairwise by selecting an instance i_R of route R leading towards a verifier as reference path and another instance i'_R of the same route as candidate path. FARs are calculated by treating all other instances $i_{\bar{R}}$ of routes $\bar{R}$ that do not lead to the same verifier as candidate paths. Candidate paths are truncated to match the temporal length of the reference path before the similarity is computed.

Both FAR and FRR drop when increasing the length of the path from one to two minutes, and reach lowest mean FAR and FRR for a duration of 2 and 6 min. Reference paths longer than two minutes exhibit mean FARs ranging between 4.3% and 7.3% while their mean FRR lies within 3.9% and 6.1%. We see a drop at $L = 2$ min, but FARs/FRRs do not change significantly beyond the that mark[11]. We thus use $L = 2$ min for the remainder of our analysis.

[11]Wilcoxon signed rank test with null hypothesis: No change in median FAR (resp. FRR) by increasing paths' temporal length. FAR: p-value 0.47, FRR: p-value 0.64. Wilcox zero treatment, sample size = 79, test statistics 182.5 and 1023.

Table 3.1: Mean and standard deviation of FAR/FRR using only the initial decision threshold the 20 paths in our dataset.

L **(min)**	**FAR**	**FRR**
1.0 min	**0.064** $\pm$ 0.057	**0.137** $\pm$ 0.143
2.0 min	**0.043** $\pm$ 0.048	**0.039** $\pm$ 0.102
3.0 min	**0.073** $\pm$ 0.085	**0.047** $\pm$ 0.107
4.0 min	**0.072** $\pm$ 0.074	**0.061** $\pm$ 0.133
5.0 min	**0.049** $\pm$ 0.061	**0.057** $\pm$ 0.103
6.0 min	**0.047** $\pm$ 0.070	**0.040** $\pm$ 0.081

3.5.7 Multi Path Learning

Multiple instances of a reference path can be used to convert the coarse initial threshold to a mixed threshold which is tailored to an individual route towards a verifier. We propose two steps to accomplish this. Firstly, by adopting a mixed threshold using Equation 3.2, a combination of d_i and d_l. Secondly, by selecting a good representative among the observed reference path instances, i.e. the medoid of the cluster.

Like the initial threshold, we set $\alpha = 0.5$ when choosing the local threshold. In case there are multiple reference paths that provide the same error rate, we use the centre-most which exhibits the smallest NW distance towards all other instances. We generate paths to estimate the distribution of between-class similarities using a Markov chain.

In order to evaluate the effects of using the medoid and Equation 3.3 for calculating the confidence factor λ, we compute the FARs and FRRs for each reference path while varying the number of instances and the decision threshold calculation scheme. Figure 3.10 shows the median FARs and FRRs for the initial, local, and mixed thresholds using 2 min reference paths. These represent doing no update, doing a full update, and a gradual update to the decision threshold, respectively. During every update, the pairwise similarities are calculated between observed reference paths, and the instance with the largest summed similarity to the other instances is selected as the medoid. This representative is used to

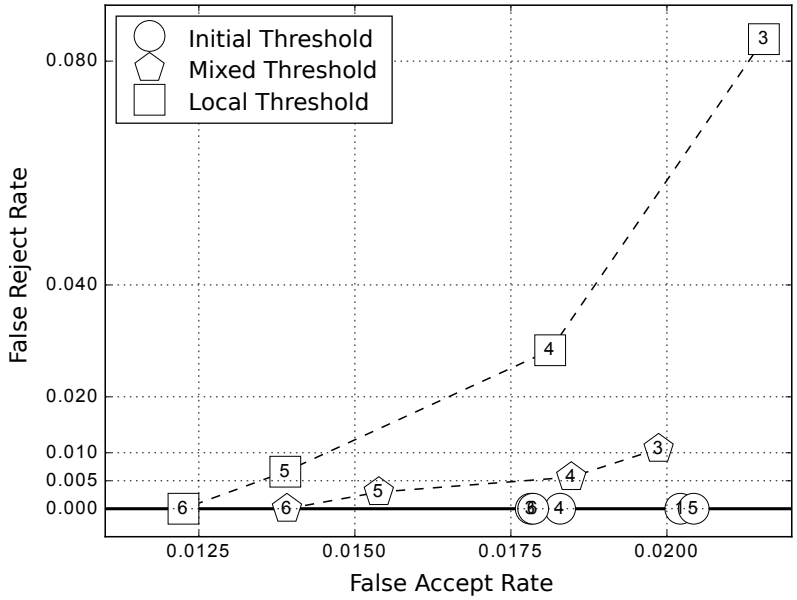


Figure 3.10: Coordinates denote median (FAR, FRR), numbers describe number of instances used. Increasing the number of observed reference path instances leads to a decrease in FARs and FRRs. The extent of this effect varies depending on the used threshold derivation algorithm.

decide whether future candidate paths are legitimate. The benefit of using the medoid to represent a group of paths is that only one similarity score needs to be calculated, and that the decision threshold can be more strict as we minimise the distance to the other instances of the group.

The circles in Figure 3.10 represent the continued usage of the initial threshold while only changing the medoid. While the median FRR remains unchanged at zero, the FAR does not improve with more instances, because the FAR depends on the decision threshold, not on the reference path representative. Using the initial threshold is equivalent to a constant confidence factor $\lambda = 0$, no trust in d_l.

The squares represents the effect of using only the local threshold. While FARs are at a similar level as the initial threshold, median FRRs are significantly worse. However, with five or more instances, median FARs drop to a lower level than with initial thresholds as threshold and medoid are more specific. Using only the local threshold is equivalent to a constant confidence factor $\lambda = 1$ (full trust in d_l).

Mixed thresholds indicated with pentagons are derived from initial and local threshold. The mixed thresholds retain the low FRR of initial thresholds after

only few reference path instances were observed, and achieve improved performance similar to local thresholds as more instances are collected.

Using the confidence factor in conjunction with the medoid to calculate decision thresholds is empirically shown to increase performance, dropping the median FAR from 2.0% to 1.5% and the median FRR from 1% to below 0.5%, when increasing the number of instances from 1 to 5. To further validate our results, we evaluate the performance of STASH when five instances are used with mixed thresholds.

Figure 3.11 shows FARs and FRRs for the 20 paths in our dataset, evaluated over different lengths, after five instances have been observed by STASH. The boundaries of the boxes denote the 75^{th} and 25^{th} percentiles, while the lines within the boxes denotes the median. The whiskers denote the range, i.e. maximum and minimum values. Overall, FARs and FRRs as well as the spread in FRRs are lower than if only the initial threshold is used (see Table 3.1). We found that FARs/FRRs for individual paths do not change significantly by considering reference paths beyond a length of 2 minutes[12]. Intuitively, the results indicate that a reference path length of 2 or 3 minutes performs better, both with respect to FAR as well as FRR in comparison to paths of the lengths 1, 4, and 5 minutes. In order to pick an actual value for an implementation of the STASH system, a user study should be conducted to find acceptable FRRs.

3.5.8 Energy Consumption

For requirement R5, an energy budget examination is necessary to understand whether STASH can be used on mobile prover devices. Hence, we conduct a controlled experiment to measure the energy consumption of our system. We obtain consumption reports after a runtime of 3 hours on three different devices[13] over ten repetitions. A second set of experiments with the same devices was conducted to control for the base consumption of the operating system. STASH was removed from the devices and all wireless connectivity was disabled during these measurements.

[12]Wilcoxon signed rank test with null hypothesis: No change in median FAR (resp. FRR) by increasing paths' temporal length. FAR: p-value 0.43, FRR: p-value 0.15. Wilcox zero treatment, sample size = 79, test statistics 779.5 and 272.

[13]Nexus 5, Nexus 5X, and Samsung GS6.

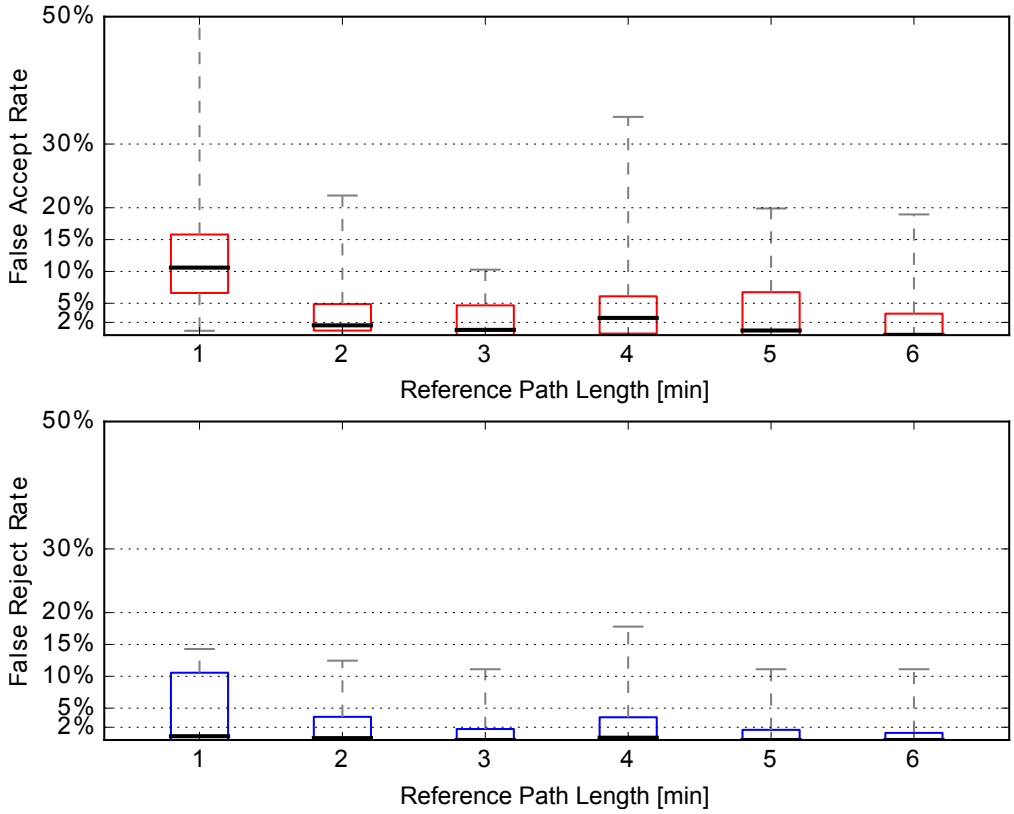


Figure 3.11: Mixed decision thresholds. Box-and-whiskers plot over FARs and FRRs of different temporal lengths after STASH has observed five instances of a reference path. On average FARs are below 3% and FRRs are below 0.5% if $L \geq 2min$.

The data is acquired using the Android Debug Bridge which provides lower and upper bounds to the battery usage at 1% granularity. We only use the upper bound estimates for this analysis to show the worst case battery drain caused by our system. All devices were fully charged before the start of each experiment. This measure was intended to control for differences in battery drainage at specific energy levels which might cause non-linear discharge behaviour [70].

Case 1: Office Worker

We model a scenario where an office employee periodically moves away from her workplace and returns after some time[14]. While she is moving, MEMS sensor data is collected at full sampling rate. When reaching the wireless transmission range of her workstation, the ZIA scheme triggers an authentication attempt between her

[14]The employee moves for 5 minutes once an hour, e.g., for a toilet, cigarette, or coffee break.

smart phone as prover and her workstation as verifier. The prover triggers path comparison every second for one minute to give the user enough time to reach the verifier. Table 3.2 reports the net contribution of STASH, compared to the control group. The increase in energy consumption varies between devices. On average we recorded an hourly drain of battery capacity by 0.56%.

Table 3.2: Net increase of hourly battery consumption in percentage of total capacity and milliampere in the office worker scenario.

Device	**Battery drain**	
	$\%/h$	mA
Nexus 5	0.41 ± 0.23	(9.32 ± 5.22)
Nexus 5X	0.63 ± 0.33	(17.13 ± 8.92)
Samsung S6	0.61 ± 0.24	(15.45 ± 6.44)

Case 2: Always Awake

We observed that the largest contributor to battery usage was the wakelock that ensured sensor measurements were processed quickly and not lost due to an overflowing sensor buffer. Therefore, we recorded the energy usage with wakelock acquired for three hours. This represents the worst case scenario. The hourly net battery drains for our devices are shown in Table 3.3. The mean increase in battery usage varies across devices, with an overall average of 1.9% per hour.

Table 3.3: Net increase of hourly battery consumption in percentage of total capacity and milliampere in the always awake scenario.

Device	**Battery drain**	
	$\%/h$	mA
Nexus 5	0.87 ± 0.28	(20.11 ± 6.40)
Nexus 5X	2.31 ± 0.24	(62.28 ± 6.60)
Samsung S6	2.55 ± 0.17	(65.14 ± 4.33)

We believe that the increased drain for the Nexus 5X and Samsung S6 is a result of their additional co-processor. This computation unit is dedicated to MEMS sensor processing while a wakelock for the inertial sensors is active.

Case 3: Daily Routine

Energy consumption of smart phone models can vary significantly for different settings and sensors. A study [71] found the average battery life to be nine hours which is equivalent to an hourly consumption rate of 11%. Assuming the power drainage over the whole capacity of a phone's battery is linear, we can estimate its effect for an employee's daily routine as follows: A user commutes for one hour and does office work for eight hours, using data from Tables 3.2 and 3.3 the additional battery drain imposed by STASH would be between 4% and 7%, depending on the phone model. These correspond to a reduction of a phone's recharge interval by approximately 20-40 minutes. In comparison, the decrease in battery lifetime when moving from an area with average WiFi to poor coverage was estimated to be 6.29% [71].

3.6 Security Analysis

The goal of STASH is to prevent relay attacks on ZIA schemes through prover-side verification of the prover device's proximity to the challenging verifier. Only if the user's current trajectory matches an authorised reference path towards that verifier, the prover device is going to grant access to pre-shared credentials and compute the authentication response.

For an adversary to launch a successful relay attack, apart from a deployed ghost device at the verifier, two conditions have to be satisfied. Firstly, the attacker is required to follow, i.e. be in wireless transmission range of the victim such that he can deploy the leech device. Secondly, the prover has to be in motion in order to produce a candidate path and that path needs to match the one associated with the targeted verifier.

As has been demonstrated by Francillon *et al.* [11], keyless-entry systems of modern cars are susceptible to relay attacks. Hence, one typical attack scenario is a criminal who tries to gain access to a car in order to steal or rob it, e.g., during the night or while the vehicle owner is at work. Since the user is usually stationary

during these occasions, no candidate path is recorded and the proximity verification fails, leading to the verifier aborting the protocol run. Thus, by design, STASH prevents relay attacks whenever the user remains stationary.

We evaluated the security guarantees of STASH while the user is moving, i.e. when a candidate path is produced, using our real-world trajectory dataset with a cumulated distance of 123 km collected over 38.6 hours of movement. We used FARs of trajectory comparison as the measure for security. We found that, on average, an adversary has a chance of less than 5% chance even if STASH has only observed one instance of a reference path (Table 3.1). This drops steadily as more instances are collected so that we can report a median FAR of 1.5% in Figure 3.10 after 5 path instances of the same route were observed.

The level of security a reference trajectory can provide is inversely proportional to the likelihood that the user follows a trajectory that does not lead to a verifier while matching a path authorised for exactly that verifier. In that case, an attacker, assuming he knows which verifier is associated with that path, could mount a successful relay attack. Based on the scenario in which STASH is deployed, the likelihood of such a false accept depends on the geographic neighbourhood or the movement patterns of the user. In this chapter, we assumed that STASH is used for premise access control where the prover is a user's personal device. Therefore, we focused the evaluation on the trajectories of an individual user by collecting a real-world dataset and a generative Markov chain to model her movement.

Yet, if the prover is a company or rental vehicle, the number of places where it could be exposed to a relay attack is much higher as it is inherently travelling more. In this case, the uniqueness of a trajectory depends on the road layout. For example, a route with many turns can be considered more complex than a straight line in most cities, but in an old town that has few straight roads, a straight trajectory may be more unique. In Chapter 5, we are going to investigate a scenario where the uniqueness of a journey through a road network is important. The evaluation of the system presented there, incorporates finding paths that are similar to a given reference path to detect relay attack opportunities.

3.7 Discussion

Transparent Authentication

The popularity of keyless-entry systems as a premium feature across different car manufacturers suggest that consumers are willing to pay for the convenience of transparent authentication. Further, Google's Trust API (Project Abacus[15]) [72] which is aimed to provide seamless user authentication on Android devices underscores zero-interaction authentication as a trend. Although STASH will prompt the user for proximity confirmation upon a failed trajectory comparison which might seem contradictory to the ZIA approach, this added inconvenience will ease over time as the system improves thresholds and accumulates new reference paths. Ultimately, STASH's contribution is to retain the convenience promised by transparent authentication while improving its resilience against relay attacks.

Reversed Path Verification

The design presented in this chapter relies on reference path instances collected over time to improve its performance. This limit's STASH to scenarios with a stationary verifier. However, since STASH represents trajectories as sequences of discrete primitives without the need to estimate absolute coordinates, these can easily be reversed. If transparent authentication for mobile verifiers is required, it is possible to compare a reference path that *starts* from the location of the verifier to a candidate path that *ends* at the verifier by inverting the reference path. For example, if a user leaves her car at a parking lot, walks away and returns later via the same route, the vehicle can act as the verifier and a smart phone or smart key as prover device.

Temporal Variability and Path Shape Sensitivity

Movement trajectories naturally contain some temporal inconsistencies between repetitions due to different movement speeds. Although we account for deviations due to sensor noise in our evaluation, one limitation of our dataset is the trajectory information which was collected over a period of one or two days per route, meaning

[15] https://www.youtube.com/watch?v=lGrRYnqHegc

that temporary obstacles like construction sites were not captured. Yet, if such obstacles persist over a longer period of time, STASH will incorporate these in its repository and start to accept the altered candidate paths. Otherwise, for one-time obstacles the user would have to manually confirm the authentication procedure.

Prover Orientation Changes

Fast changes in the prover's orientation can negatively affect the performance of our system. For example, in case a smart phone is used, taking the phone out of the pocket disturbs the gravity data required for turn detection. During our experiments, we avoided this by attaching the prover to a bike. In general, vehicles such as wheelchairs or cars provide more reliable gravity estimates, thus present an attractive use case for STASH deployment.

Movement Primitives

While we currently consider movement as a binary category, STASH can easily be extended to operate on a more complex primitive. For example, using a transport mode detection scheme [73], trajectories could be made more unique depending on the different means of transportation used by a user. As such, a path that includes cycling, then taking a tram, and then walking would be more specific than if all movement intervals are represented by the same primitive.

3.8 Summary

In this chapter, we presented how ZIA schemes implicitly rely on the location of the user through the proximity assumption between prover and verifier. We argued that proximity confirmation by the verifier alone is not sufficient to prevent relay attacks as common methods that relay on measuring the RSSI of received packets or their ToF would measure the packets exchanged with the attacker's ghost device.

We proposed to use the way a user approaches a verifier device as evidence for the prover that both devices are indeed co-present. In several ZIA application scenarios, a user's movement is an inherent repetitive pattern since she first has

to travel to the location where authentication is required. Once the prover has confirmed the validity of the authentication request, it participates in a CHAP by computing a response using a secret established during an enrolment procedure. Since STASH is entirely realised on the prover, it allows easy integration into existing systems while preserving the user's privacy at the same time.

After presenting the design of STASH we conducted a performance evaluation using both, from a generative Markov chain as well as experimental data. For the latter, we implemented a prototype for smart phones running the Android operating system and collected trajectory data from 20 different routes in two cities over 38.6 hours of movement with a cumulated distance of 123 km. During this evaluation we show where the system either fulfils requirements R1-R5 or offers adjustable parameters to customise its behaviour. With modest resource requirements, STASH provides a valuable extension for existing ZIA schemes.

While it is sufficient to verify the co-presence of a stationary verifier and a mobile prover in situations when user authentication is required, a different more dynamic application domain, vehicular networks, requires co-presence verification of moving vehicles. In the next chapter we review state-of-the-art message authentication schemes as well as common privacy and security requirements for VANETs. We discuss the dependence of VANETs on location information before extending the idea of a device's movement to be used as an indicator for proximity to the scenario of multiple vehicles.

In the realm of ideas everything depends on enthusiasm... in the real world all rests on perseverance.

— Johann Wolfgang von Goethe

4

Security and Privacy in Vehicular Ad-hoc Networks

Contents

4.1 Introduction 58
4.2 Vehicular Communication 59
4.3 Applications 60
4.3.1 Safety 61
4.3.2 Efficiency 62
4.4 Requirements 63
4.4.1 Authenticity 64
4.4.2 Integrity 64
4.4.3 Non-repudiation 64
4.4.4 Conditional Anonymity 65
4.4.5 Revocation 65
4.5 Message Security 65
4.5.1 Identity-based Cryptography 66
4.5.2 Group Signatures 66
4.5.3 Symmetric Cryptography 67
4.5.4 Asymmetric Cryptography 67
4.6 Passenger Privacy 68
4.7 Pseudonym Change Schemes 71
4.7.1 Silent Periods 72
4.7.2 Encrypted Mix-Zones 73
4.8 Summary 74

In the previous chapter we analysed how proximity information is implicitly

used to ensure the co-presence of a mobile and a stationary device when performing wireless ZIA. We presented how this assumption can be overthrown by an adversary with Dolev-Yao capabilities who deploys a wireless relay. In this chapter, we are going to provide a background of inter vehicle communication and its applications to improve several aspects of modern transportation. Afterwards, we review the requirements for VANETs and the associated security as well as privacy implications of information sharing in this highly location dependent setting. Finally, the *de facto* architecture for an intelligent transportation infrastructure is presented.

4.1 Introduction

Although governmental support for public and mass transportation has increased, cars still remain the dominant mode of transportation. Despite drawbacks in safety [74], efficiency [75], and sustainability [76], their flexibility and availability is unmatched by any other means of transportation [77]. This persistent use of motorised vehicles, combined with the emergence of smart cities as a new area of research has created a renewed interest in the field of VANETs, a Mobile Ad-hoc Network (MANET) derivative first mentioned in 2001. Moreover, recent advances in sensing technology such as LIDAR promise to increase the capabilities of individual vehicles promoting a new push towards autonomous driving, a concept with its roots dating back to the 1980s.

After combining these two concepts, connected autonomous vehicles are expected to play a key role in the future of Intelligent Transportation Systems (ITSs), which draw on collaborative approaches to increase the efficiency and safety of our roads. In effect, large-scale field trials are currently being conducted by major high-tech companies to explore the capabilities and limitations of vehicular communication [78–81]. Their first results led to the development of safety and efficiency enhancing applications. These methods leverage location based information which is collectively gathered and shared between vehicles using a VANET.

4.2 Vehicular Communication

The main groups working on vehicular communication are the ETSI Technical Committee on ITS[1] [82] and the ISO Technical Committee 204[2]. Their goal is to design an open communication network that facilitates the exchange of relevant information between traffic participants. The main entities of a VANET are vehicles equipped with On-Board Units (OBUs) and stationary Road-Side Units (RSUs) positioned along the roadway. While OBUs enable the wireless exchange of information for vehicles, RSUs act as an auxiliary infrastructure to support data delivery at busy areas of the road network, e.g. intersections, and provide connectivity with infrastructure services to access weather data or directions.

Figure 4.1 provides an overview of VANET communications which are commonly categorised into Vehicle-to-Infrastructure (V2I) and Vehicle-to-Vehicle (V2V) messages[3]. Although current research mostly considers OBUs and RSUs as communication partners, supporting a wider range of receivers is already envisioned. For example, portable devices like smart phones could provide the location of particularly vulnerable road users such as motorbikes, bicycles, and pedestrians. Thus, all transmissions involving a vehicle are grouped under the term Vehicle-to-X (V2X) communication.

The two dominant approaches for information dissemination in VANETs are via a cellular data link based on Long-Term Evolution (LTE) 5G [83] or using Dedicated Short Range Communication (DSRC) based on the IEEE 802.11p communication standard [84]. While for convenience functions and status updates LTE is sufficient, DSRC was designed as a connectionless protocol with low latency to support safety critical applications with real-time requirements. IEEE 802.11p belongs to the 802 WiFi standards family and has channels allocated in the frequency range of 5.85-5.925 GHz. The idea behind selecting frequencies close to the 5G WiFi standards

[1] https://portal.etsi.org/TBSiteMap/ITS/ITSToR

[2] https://www.iso.org/committee/54706.html

[3] Source for Figure 4.1: https://sites.google.com/a/ochin.mygbiz.com/my-scientific-work/my-academic-work/openvanet

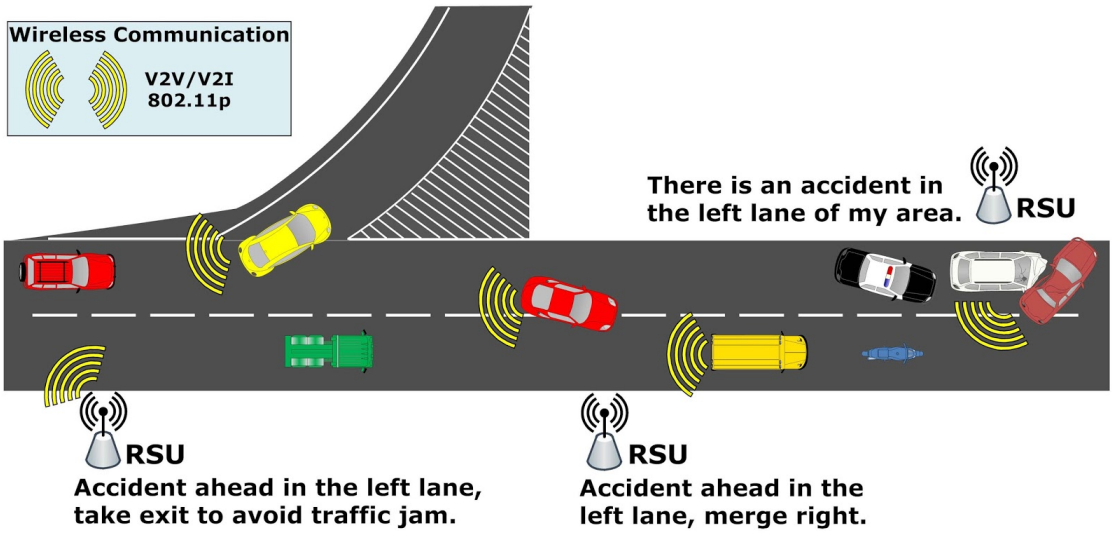


Figure 4.1: The entities in a VANET are RSUs and OBUs. Using the WiFi-like IEEE 802.11p standard, location based information is shared via V2V and V2I communication.

was to enable an easy adoption by simply extending existing WiFi chipsets found in modern mobile devices such as smart phones, tablets, and laptops.

Since the IEEE 802.11p standard relies on broadcast communication without dedicated infrastructure, e.g. access points, the transmission range of an individual vehicle is limited by its transmission power and obstacles, such as trees, buildings, and other vehicles. Using industrial grade hardware, a single vehicle can cover an area of up to 300 meters in diameter. A common technique to overcome this limitation is to implement peer-to-peer communication that passes information over a larger distance by re-broadcasting messages using hop-to-hop routing algorithms. This means that messages are temporarily carried by other network nodes and then either released at their destination or handed over to the next hop in the delivery chain.

4.3 Applications

So far, research of VANET enabled applications has focused on two topics: safety and efficiency. Both types leverage the situational awareness provided through beacon messages, called Cooperative Awareness Messages (CAMs) in the European standard [85] and Basic Safety Messages (BSMs) in the United States [86]. A vehicle's OBU includes its exact location, heading, and velocity in these messages

that are broadcast at a frequency of 1-10 Hz. So created knowledge about the local traffic allows vehicles to react faster, foresee velocity changes or even accidents, and *look* further ahead than a human would be able to.

4.3.1 Safety

Despite the overall decline in road deaths, the real progress does not yet meet the target set by the European Commission when it comes to road safety[4]. We are now going to illustrate how VANET safety applications are supposed to help reaching this goal using three examples: Electronic Emergency Brake Light (EEBL), Local Danger Warning (LDW), and Cooperative Collision Avoidance (CCA).

Electronic Emergency Brake Light

When a vehicle performs emergency braking, its OBU issues a report if the experienced deceleration reaches a certain threshold [14]. Using multi-hop communication, this report propagates to following vehicles which then warn the driver or actively lower vehicle speed in order to prevent rear-end collisions [87].

Local Danger Warning

Inter vehicle communication can help to increase the driver's horizon by sharing the driving status and intentions between vehicles in their vicinity [15]. For example, a truck can provide information to following vehicles that allow them to *see through* it in order to choose a good time to overtake safely. Similarly, at intersections where roads are obfuscated through buildings, vehicles can relay information to let incoming vehicles *look around the corner*. Localised danger warnings can also indicate vulnerable road users like cyclists or pedestrians which might be located in a driver's blind spot.

[4] https://etsc.eu/euroadsafetydata/

Cooperative Collision Avoidance

In order to avoid collisions, collaborative algorithms have been developed which compute, using the information of beacon messages, a set of states that the vehicles involved must not enter [88]. From these unsafe states, feedback maps are computed that capture the state transitions that could lead to collisions. Afterwards, the control inputs of each vehicle will be constraint such that these state changes cannot be brought about by the driver [89].

4.3.2 Efficiency

Situational awareness can also be used to improve the financial and ecological impact of using cars for transportation. While Cooperative Adaptive Cruise Control (CACC) aims to pack vehicles tighter on the road to increase transportation efficiency, Virtual Traffic Lights (VTLs) try to reduce infrastructure spendings and overall time spent on the road.

Cooperative Adaptive Cruise Control

Vehicle platoons are groups of vehicles that drive in a formation and rely on CACC to control the acceleration of member vehicles. The advantages of such platoons are manifold: A decrease in fuel consumption through a reduction in air resistance (up to 10% for trailing vehicles [90]) comes with decreased environmental pollution [91, 92]. In 2016, the European Union organised the European platooning challenge with participants like Daimler, MAN, and Volvo to promote these efforts. In the future, following vehicles might even delegate their steering to autonomously driving platoon leaders to save manpower [93]. Platoons also promise a more efficient utilisation of the road infrastructure and fewer congestion [94]. Field trials for highway based platoons have returned positive results [17] and current mobility research tries to transfer the advantages of platoons into the urban scenario [16].

Virtual Traffic Lights

According to their original design, traffic lights followed a static schedule that allocates a certain green period to each road at an intersection. One modernisation of this architecture was the integration of induction loop into the tarmac. Using these sensors, traffic lights were able to determine which roads are empty and which have a car waiting to pass the intersection. In a new attempt to further improve traffic signalling, the idea of replacing physical traffic lights with virtual ones emerged [95]. Using V2V communication, the first vehicle that arrives at an intersection is elected as manager and computes a schedule for all other vehicles to pass the junction.

4.4 Requirements

As discussed in the previous section, information disseminated through the VANET will eventually be integrated with technologies like self-driving cars and driver assistance systems. Given the human involvement in such systems, driving safety is of paramount importance and requires the communication protocol to implement sufficient security measures. Otherwise, adversaries can cause accidents leading to physical harm or even the death of passengers. For example, an attacker can generate misleading messages to disable CCA [96] or other safety applications like EEBL and LDW [18–20]. Therefore, a mechanism to ensure integrity, authenticity, and non-repudiation of V2X messages is needed.

On the contrary, messages containing location information, such as mandatory CAMs allow to track vehicles [21]. Consecutive messages from the same vehicle can leak information such as its itinerary or even the identity of the driver [22]. Hence, it is necessary for message signatures to not carry identifying information and drivers can remain anonymous. In IEEE standard 1609.2 [97], secure message formats and processing procedures for information used in VANETs are specified. We are first going to describe the requirements found in VANETs, before we review how different message signature schemes aim to fulfil these, in the next section.

4.4.1 Authenticity

Although the VANET was designed as an open network with respect to receiving messages, vehicles have to provide verifiable signatures if they want other vehicles to accept their messages. Further, authentication allows to control the authorisation level of vehicles, e.g. a consumer vehicle must not be able to send emergency beacons emitted by an ambulance truck [98]. Additionally, authenticated messages are bound to a single identity, thus one vehicle alone cannot pose as multiple cars to create a non-existing congestion in a Sybil attack [99].

4.4.2 Integrity

Integrity mechanisms prevent the injection of forged CAMs and the alteration of VANET messages in transit. A common way to guarantee data integrity is to use a one-way hash function to generate a message authentication code. Assuming that the function is truly one-way and different inputs do not result in the same output, i.e. a collision, the recipient of a message can verify its contents by computing and then comparing the message authentication code [100]. One method to ensure integrity and authenticity at the same time are Hashed-Message Authentication Codes (HMACs).

4.4.3 Non-repudiation

Traffic violations are traditionally prosecuted by law enforcement according to evidence from witness testimonies, recordings of cameras, or other means of surveillance. A VANET message can either be used as evidence to directly convict an offender, e.g. using CAMs to recognise speeding, or indirectly, e.g. from a black box of a vehicle that relied on data received from a malicious vehicle. In both cases it is necessary that the sender cannot dispute the origin of the message. Similarly to preventing impersonation, non-repudiation is commonly achieved by irreversibly tying an identity to a message using a digital signature [101].

4.4.4 Conditional Anonymity

In order to protect the privacy of passengers, a vehicle should remain anonymous to other VANET participants when sending messages. At the same time, every vehicle must use an uniquely identifiable signature to ensure that law enforcement authorities are able to retrieve the real identity of the driver in case misbehaviour is reported. This requirement is usually resolved on an architectural level by requiring vehicles to obtain signing material from a globally trusted party which retains a mapping to the real identity of the vehicle owner. For privacy protection, several approaches suggest a separation of the issuing entity and the identity resolution authority [102, 103].

4.4.5 Revocation

If a driver has been found guilty of misbehaving or a vehicle was infected by malware and has to excluded from the network, its signature generation material has to be invalidated. After this revocation process, messages emitted by the concerned vehicle are no longer accepted by other VANET entities. Approaches to implement revocation can be divided into two categories: passive and active revocation. Passive revocation relies on short validity periods included in signing material which requires vehicles to request new credentials periodically [104]. During each request, the issuing authority can check whether the identity of the requester has been revoked. Although this technique mitigates scalability issues, it introduces delays as it can only revoke credentials at specific timeslots. Active revocation on the contrary uses Certificate Revocation Lists (CRLs) that are pushed to all nodes in the VANET to facilitate a more timely exclusion of a node from the network [105].

4.5 Message Security

A wide variety of schemes have been proposed to address aforementioned requirements. All approaches produce some form of digital signature using vehicle-specific signing material. A message's signature must provide verifiable authenticity and integrity of the payload as well as guarantee non-repudiation. Further, it

must be impractical for an attacker to forge a signature without possessing valid VANET credentials. In order to provide conditional anonymity and revocation, these approaches differ depending on the underlying cryptographic primitives. In the following, we are going to briefly characterise the four most established approaches: identity-based cryptography, group signatures, symmetric cryptography, and asymmetric cryptography.

4.5.1 Identity-based Cryptography

Approaches incorporating Identity-Based Cryptography (IBC) were deemed attractive due to their small resource footprint. By enabling signature verification using a vehicle's identity as its public key [106], the need to exchange a sender's key and certificate is eliminated [107]. Using bilinear pairing, two vehicles with a private key obtained from a mutually trusted authority can verify their message signatures without revealing their identities [108]. However, as shown in [109], a sender can void accountability unless a zero-knowledge proof is performed when signatures are verified. This additional overhead is yet to be considered to evaluate whether its performance makes IBC signatures feasible for VANETs. Additionally, in order to evict malicious vehicles, the trusted authority has to assign and distribute new keys to all vehicles [110] which is arguably the biggest disadvantage of IBC schemes.

4.5.2 Group Signatures

As a decentralised approach, group signatures organise vehicles around a group leader who provides each vehicle with a separate private key [111]. All signatures generated under these keys are indistinguishable and can be verified using the group's public key, thus provide an anonymity set with the size of the group. It is the group manager's responsibility to hold a mapping between keys and identities to enable non-repudiation and revocation [112]. Although such schemes scale better since revocation only affects a reasonable number of vehicles, a group manager needs to be elected and takes the role of a trusted party who could impersonate

any of the group members. While drivers might be willing to trust a governmental institution this might not be the case for an arbitrary vehicle on the road.

4.5.3 Symmetric Cryptography

Symmetric cryptography based schemes use HMACs which change dynamically with a message's content, thus are anonymous to an observer. The required keys are provided by a trusted signature verifier, e.g. a roadside unit [113]. The RSU maintains a mapping between vehicles and short-term keys for non-repudiation and revocation. Although such approaches allow significantly more efficient signature generation and verification [114], they introduce additional complexity with respect to key management and signature verification. Since symmetric keys must not be disclosed with other vehicles, verification can only be performed by the key distributor. Therefore, the two approaches in [113, 115] and [114] either require a vehicle to forward messages to RSUs for verification or RSUs to periodically broadcast a set of verified HMACs. In order to mitigate this drawback, the work in [116] proposed to use a hash chain mechanism with delayed key disclosure. Unfortunately, the mechanism increases an attacker's ability to track vehicles [117].

4.5.4 Asymmetric Cryptography

The Elliptic Curve Digital Signature Algorithm (ECDSA) [118] has received a lot of attention and is currently the most likely signature scheme to be adopted by standardisation bodies in Europe and the United States (IEEE 1609.2 WG [119] and ETSI [120]). This signature generation algorithm requires Elliptic Curve Cryptography (ECC) credentials in the form of a private/public key pair. A sender uses its private key to compute a signature over a message both of which are transmitted along with a certificate including the sender's public key. When receiving a message, the public key can be used to validate the attached signature while the validity of the public key itself is confirmed using a Public-Key Infrastructure (PKI) maintained through RSUs [121]. The PKI derivative that is optimised for the use in VANETs is commonly referred to as Vehicular Public-Key Infrastructure (VPKI).

Digital signatures using public key cryptography ensure the authenticity, integrity, and non-repudiation of messages. Additionally, to protect a driver's privacy, signing credentials are anonymised to decouple them from the real identity of the driver [117].

As specified in the respective standard, each OBU, has to be authorised before it can participate in a VANET and provided with individualised signing material. Schemes like SECMACE [122, 123] provide a VPKI consisting of several Certificate Authorities (CAs) for credential provisioning. Since these CAs bear a lot of responsibility and require to be trusted by all drivers, governmental institutions were suggested to fulfil this role. For example, when the owner of a vehicle registers for a license plate, the same organisation could simultaneously issue VANET credentials.

The widely accepted architecture to manage pseudonym issuance is shown in Figure 4.2. It features three types of CAs, Long Term CAs (LTCAs) and Pseudonym CAs (PCAs) both are trusted via a global Root CA (RCA). Due to differences in legislation and vehicle registration between countries, the VPKI can be separated into domains where each country operates its own LTCA [124]. The LTCA provides registered vehicles with a Long Term Certificate (LTC) which is used to authorise the acquisition of pseudonyms from a PCA[5]. Common assumptions are that the RCA's certificate to verify the LTCA's and PCA's identities is installed on all VANET entities, i.e. OBUs and RSUs, and VPKI communication is protected using TLS. In case a faulty behaviour is detected, authenticated messages can be used by a Resolution Authority (RA) to retrieve the long-term identity of the sender vehicle [123]. The misbehaving OBU is subsequently excluded from further participation in the system by invalidating its credentials that are added to a revocation list [105].

4.6 Passenger Privacy

The signature generation schemes we reviewed earlier rely on two mechanisms for privacy protection: cryptography or proxy identities to provide sender anonymity.

[5]Without the loss of generality, we omit the separation into multiple LTCAs and PCAs according to a vehicle's home domain for the remainder of this dissertation. We refer the reader to Khodaei *et al.* [122] for an illustration how this separation can be implemented.

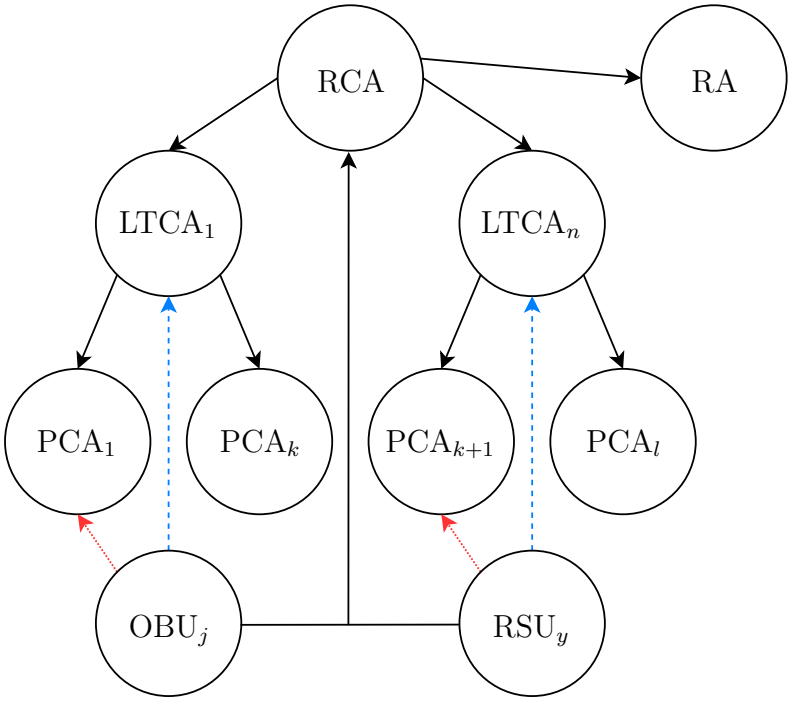


Figure 4.2: Certification hierarchy of the VPKI. Solid black arrows indicate trust relations, dashed lines show pseudonym requests, and red dotted lines highlight pseudonym acquisition.

Identity-based and symmetric cryptography [106] based schemes fall into the former, while public key cryptography [125] belong to the latter category. The credentials used in asymmetric cryptography based schemes are commonly referred to as *pseudonyms*. For the remainder of this dissertation, we will be concerned with these schemes since they are more likely to be put into practice.

Due to the broadcast nature of V2X communication, messages can be received by anyone within a 300 meter perimeter [23, 24, 126]. Since these messages are signed with a vehicle's private key, both actions and locations can be linked to the owner's digital identity. While pseudonym-based message authentication schemes allow a trusted authority to resolve message signatures to a driver's identity for lawful prosecution, they provide theoretical anonymity against eavesdropping on the network [117]. However, the continuous monitoring of messages allows vehicle tracking by linking messages to the same sender [127]. Furthermore, even though the owner's identity is not directly exposed in vehicular communication, it was shown that a driver's home and work address can be inferred using the

start and end location of trips which ultimately can lead to the identification of the driver herself [22–24, 126].

As the majority of commuters travels within the same city or arrive from nearby suburbs, it can be assumed that the maintenance of the road infrastructure is done by a single organisation. This allows the theoretical collection of VANET messages which provide enough information to compromise location privacy and hence the driver's identity. Adversaries who have the potential to perform such attacks can be governmental institutions or road infrastructure providers [128].

In their survey, Petit *et al.* [117] categorise attacks that aim to compromise passenger privacy along three dimensions. These reflect different capabilities of collecting and generating of VANET messages as well as interacting with network entities.

Local or Global: This defines the coverage capability of the attacker. The local adversary controls a small number of eavesdropper devices that cover a subset of the deployed VANET, e.g. several intersections. A global attacker on the other hand is able to receive all transmitted messages.

External or Internal: An internal adversary is in possession of valid VANET credentials which allows him to produce messages that are accepted by other participants. On the contrary, the external attacker is subject to intrusion detection systems and hence has a limited range of possible attacks.

Active or Passive: Passive and active adversaries differ in the sense that the former simply eavesdrops on the communication while the latter can try to modify or rebroadcast messages.

While Petit *et al.* [117, 129] limit their analysis to local adversaries, we argue that due to the design of 802.11p as an open network close to the WiFi spectrum, it is only a matter of time until VANET messages have to be considered widely and publicly available. For example, a crowd sourced listening infrastructure exists for aviation [130] as well as a myriad of websites that provide real-time information of air planes. Additionally, given the availability of cheap SDRs, a network of listening posts that covers a city does not seem farfetched [131].

Independent of the observation capabilities of an attacker, schemes that treat pseudonyms as ephemeral rather than static credentials were proposed to mitigate the risk of privacy compromise [132, 133]. Each vehicle is provided with multiple instead of only one digital identity which can be used interchangeably to sign VANET messages. By frequently changing to a new pseudonym, a journey is broken down into smaller, unlinkable segments. At the same time, identifying material, such as Media Access Control (MAC), Internet Protocol (IP) addresses, and other intrinsic communication stack data, has to be changed to prevent re-identification of a vehicle [134, 135]. Lastly, since OBUs may have simultaneous access to multiple identities, mechanisms to prevent Sybil attacks have to be taken into account when designing pseudonym change schemes.

We note that in some countries like the United Kingdom, number-plate readers are used to track vehicles and provide information vital for law enforcement. Pseudonym schemes do not interfere with these capabilities, since credentials issuing authorities are still able to track vehicles, but rather aim to prevent external adversaries from re-purposing VANET identifiers to compromise passenger privacy.

4.7 Pseudonym Change Schemes

Figure 4.3 shows the basic principle underlying any pseudonym change scheme. Instead of using the same singing material over the whole journey, as done by OBU_0, a trip is divided into smaller segments by changing from one pseudonym to a new pseudonym as indicated for OBU_1. Given that a vehicle must not reuse a pseudonym to prevent re-identification, the idea behind the segmentation is that it is more difficult for an attacker to reconstruct the whole journey of a driver, thus minimising the surface for privacy compromise.

However, it has been understood that messages signed under different pseudonyms can be linked by an eavesdropper to compromise pseudonym transitions and defeat the pseudonym change scheme; either by examining the pseudonyms themselves or by basing inferences on the content of the signed messages [127]. Such attack are called *syntactic* or *semantic* linking [127, 136] both of which exploit the circumstances

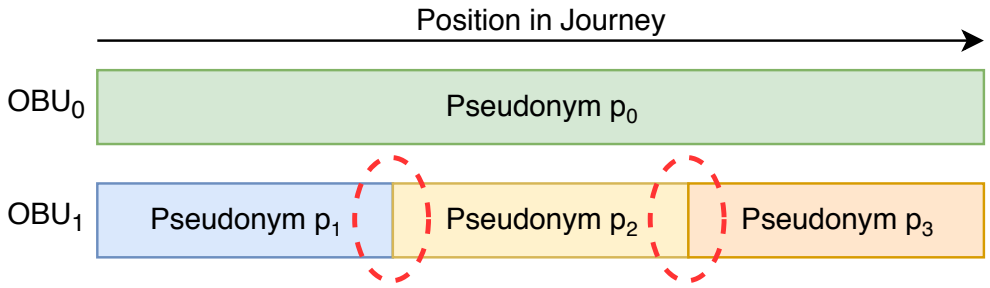


Figure 4.3: Two vehicles with different pseudonym strategies. While OBU_0 signs messages with the same pseudonym for its entire journey, OBU_1 uses three pseudonyms to separate its trip into segments. Transitions between pseudonyms are highlighted with red dashed circles.

under which vehicles change pseudonyms. Based on the time of a transition, an attacker might observe an isolated pseudonym change, and associate the old and new identifiers through *syntactic linking* [136, 137]. During a *semantic linking* attack, the adversary uses physical constraints of the road layout, velocity, and heading of a victim's vehicle to predict its trajectory and link consecutive messages [21, 127]. This information is available through the mandatory CAMs broadcast by all vehicles.

While appropriate pseudonym provisioning policies can alleviate syntactic linking through transition synchronisation [122, 123, 125, 138], privacy protection can still be lowered by semantic linking [21, 127, 136, 139, 140][6]. Different pseudonym transition strategies have been proposed to withdraw the basis for semantic linking attacks. These either use silent or encrypted periods and differ mostly in how the time and location for such periods is chosen.

4.7.1 Silent Periods

Inspired by the work of Huang *et al.* [142], who described the use of silent periods against honest-but-curious service provides, at first CARAVAN [143] and later AMOEBA [128] were proposed. Both approaches suggest that each vehicle should turn its wireless transmitter off for a randomly chosen interval and change pseudonym within that silent period. Despite the positive effect on privacy protection, both

[6]Note that connecting such anonymous location profiles to real identities of vehicle owners is the final step, e.g., tracing their commutes and identify home/work locations [141], or full de-anonymization of vehicles by *honest-but-curious* VPKI entities [122, 123, 137].

systems imposed a performance penalty on safety applications, switching CAMs off for more than 2 s can lead to physical harm for passengers [144]. As a mitigation, vehicles could change their pseudonyms when their speed drops below 30 km/h [136]. The scheme called SLOW uses vehicle density as contextual parameter to trigger silence periods only when vehicles are moving slowly. Even though this technique has a smaller impact on safety applications, it was shown that an adversary can still conduct syntactic linking due to a lack of synchronisation [137] and semantic linking by predicting their trajectories [145]. Additionally, its reliance on congestion and low vehicle speed for pseudonym change limits its applicability under low traffic density conditions or on highways.

4.7.2 Encrypted Mix-Zones

Another line of research which does not impair safety applications proposes encrypted pseudonym transitions. In Cryptographic mix-zone (CMIX) based schemes [146, 147], a cryptographic mix-zone is created at appropriate times and places, e.g. at intersections. When crossing these regions, vehicles change their pseudonyms privately while their communication is encrypted which prevents syntactic and semantic linking attacks. However, the achieved privacy protection highly depends on the number of vehicles participating in the mix-zone. Consequently, user privacy is degraded under low traffic density, e.g. in a highway scenario [129]. Moreover, an attacker could compromise pseudonym changes within a mix-zone based on the traffic mobility pattern and vehicle speed [148].

An alternative to these static mix-zones, dynamic formation of mix-zones were proposed [149]. Each OBU is provided with a global symmetric key, which it uses to spontaneously initiate an encrypted area for pseudonym change. To prevent semantic linking, such an approach requires vehicles to change their speed, lane, or direction when updating their pseudonyms, bringing the practicality of this scheme into question. Since the scheme allows any OBU to terminate encryption periods, an internal attacker could use this to reduce the anonymity set, thus compromising user privacy. Additionally, the scheme introduces significant overhead

on key management; for each revocation event, a new symmetric key has to be distributed to all legitimate vehicles within the system.

4.8 Summary

We reviewed the current state of VANET research by listing its potential applications followed by the requirements on vehicular communication. Afterwards, we discussed the security and privacy solutions proposed to enable the development of safe intelligent transportation systems.

Considering the risks of relying on location information in mobile systems described in Chapter 2, we set the stage for the coming chapters. In the next chapter, we investigate a method to verify the co-presence of two vehicles in a privacy preserving way. Thereafter, a method to improve privacy without diminishing the safety gains of VANETs is presented in Chapter 6. And finally, in Chapter 7, a message authentication scheme that fulfils all VANET security and privacy requirements while providing driver-centric accountability is proposed.

A great wind is blowing, and that gives you either imagination or a headache.

— Catherine the Great

5

Co-presence Verification in Vehicle Formations Using Driving Trajectories

Contents

5.1 Continuous Co-presence in Vehicle Formations 76
5.2 System Proposal 77
5.3 Concept and Assumptions 79
5.3.1 Application Scenario: Formation Co-presence 79
5.3.2 System Model 81
5.3.3 Adversary Model 83
5.3.4 Design Goals 84
5.4 Get-In-Line Architecture 86
5.4.1 Trajectory Representation 86
5.4.2 Intended Trajectory 87
5.4.3 Past Trajectory 89
5.4.4 Co-presence Verification Protocol 91
5.5 Evaluation 95
5.5.1 Map Analysis 96
5.5.2 Security Analysis 101
5.5.3 Experiments 106
5.6 Discussion 109
5.7 Summary 110

As presented in the previous chapter, data security of VANET messages can be guaranteed using appropriate cryptographic mechanisms. Yet, inter-vehicle communication is inherently location dependent and physical properties like spatial

proximity and driving direction are not considered in the security standard IEEE 1609.2. For example, if an ambulance requires an emergency corridor or a car recognises a hazardous situation like black ice, it is important to propagate this information to the surrounding vehicles. Recipients determine the relevance of messages either directly using GPS data included in the payload (CAMs, etc.) or indirectly through routing information which addresses a certain geographical area. Yet, GPS information can be unavailable due to obstacles blocking the line-of-sight between receiver and satellites or is not necessarily trustworthy, as discussed in Chapter 2.

Especially, since the reliability of VANET messages is essential for safety applications, it is necessary to verify the location of sender and receiver. In this chapter we investigate a system that uses a vehicle's trajectory to confirm location claims of co-present vehicles in a privacy preserving way. Our method draws on the insights gained in Chapter 3 where we used a user's trajectory to establish the proximity of a prover and a verifier. We extend this idea and use a vehicle's trajectory through a road network to uniquely define its location as well as its driving direction. This chapter is based on our work in [150].

5.1 Continuous Co-presence in Vehicle Formations

As described in Section 4.3, vehicle platoons are formations that consist of a dedicated platoon leader and several platoon members driving in succession. Leader vehicles are actively managing the movement of platoon members using VANET messages, thus it is necessary to ensure that the communicating parties are indeed co-located vehicles. Since platoons are inherently of dynamic nature [151], new vehicles should be able to automatically join and leave. To facilitate this, a platoon leader is assumed to announce its presence to nearby vehicles allowing them to become part of the group upon request. When receiving a join request the platoon leader has to verify the location of the requesting vehicle and only accepts if it is in fact part of the platoon formation.

The generic attacks on VANETs described by Han *et al.* [51] and Bißmeyer *et al.* [19] also apply to vehicle platoons. In these attacks non-existing ghost vehicles are inserted into the traffic, pretending to be co-located to the victim. For the platoon scenario this means that an attacker could be recognised as a formation member through a forged join request. Hence he can broadcast messages that manipulate the inputs used by the platoon leader to control other vehicles, i.e. by emulating a braking or accelerating vehicle inside the platoon. In their study on vehicular platoon misbehaviour, DeBruhl *et al.* [18] identify several attack strategies: mis-report, collision induction, and reduced headway. In fact, small fluctuations in acceleration can amplify along a platoon reducing its string stability [152], and may ultimately lead to crashes and fatalities.

Existing solutions to verify the co-presence of vehicles use specialised antennas which allow to place a sender in front or behind of the receiver but do not offer an estimate of its distance [93, 153, 154]. A combination of directional antennas and distance bounding was proposed in [20] and proven effective in highway simulations. However, due to the road density in urban environments and a slower driving speed, their reliance on wireless transmission range for distance bounding introduces inaccuracies. As a result, a receiver cannot determine whether a sender is on the same, an orthogonal, or a parallel road. Moreover, a relaying attacker can add a delay to place a sender further away from the receiver than she actually is. Lastly, the approach also relies on majority vote which makes it susceptible to collaborating attackers or an individual attacker with sufficient resources. It also requires re-authentication whenever a vehicle changes identity and hence cannot be combined with pseudonym schemes without reducing their efficiency.

5.2 System Proposal

While the idea of using unlinkable pseudonyms is wildly advocated to provide passenger privacy, applications like CACC require exactly this linkability to track the whereabouts of co-present vehicles [155]. Therefore, a technique that allows vehicles to reliably validate location claims despite the presence of pseudonym

schemes is needed. In this chapter, we propose Get-In-Line, a system that supports these privacy requirements and provides linkability *within* a vehicle formation while preserving unlinkable for an *outside* observer. We present a method for ongoing co-presence verification in a privacy preserving way. It enables vehicles to distinguish between messages from vehicles travelling in the same direction alongside them and messages from vehicles that are stationary or moving in a different direction. The presented scheme first authenticates vehicles as they start following each other, and then de-authenticates them once their paths deviate. We implemented a smart phone application to evaluate the performance and feasibility of our system using real-world experiments. The security of the protocol is conceptually analysed and tested through realistic traffic simulations. We summarise the contributions made in this chapter as follows:

- We present a novel approach to verify the ongoing co-presence of vehicles in an urban environment. We exploit the characteristics of a trajectory through a road network and require a vehicle to share the same route as a leading vehicle in order to become a verified following vehicle. Co-present vehicles gain knowledge of verified neighbours, thus the capability to authenticate their VANET messages. This allows authenticity checks for safety critical applications.

- We design a protocol that verifies the ongoing co-presence of two vehicles. The construction only reveals a driver's trajectory to other co-present vehicles and hence protects passenger privacy against an eavesdropping attacker. Our approach operates transparent to pseudonym schemes and thus cannot be exploited to perform linking attacks by attributing different messages to the same sender.

- Using an Android application, we conduct experiments with two cars to evaluate the performance of our scheme. Furthermore, we conceptually analyse its security properties and assess its performance through realistic traffic simulations. We use FARs and FRRs to summarise our findings.

5.3 Concept and Assumptions

For our scheme, we envision different stages in the process of confirming a vehicle's co-presence. In this section, we start with an application scenario to explain these stages and how they naturally occur while vehicles traverse a road network. After presenting the system and adversary model, we follow with a discussion of the system requirements.

5.3.1 Application Scenario: Formation Co-presence

Similar to platoon systems, we exploit vehicles driving in succession and use their trajectory through a road network to verify ongoing co-presence. Based on this method, a Leader Vehicle (LV) can promote unverified Candidate Vehicles (CVs) to verified Follower Vehicles (FVs). Even though we only consider spontaneous platoon formation, complimentary approaches such as PrivatePool [156] could be used to negotiate rendezvous between vehicles and increase efficiency.

The application scenario for our system is illustrated in Figure 5.1. The vehicle formation consists of the leader vehicle LV and several following vehicles FV, with their route indicated as a dashed line. A candidate vehicle CV which approaches from a different route, highlighted with a dotted line, eventually intersects with the formation's route. Until this point, the CV's ongoing co-presence is not confirmed, hence it does not belong to the set of verified following vehicles.

Once the CV senses the formation's proximity through periodically broadcast announcements sent by the LV, the system is triggered. These beacon messages include the leader's location to indicate its presence. When the CV meets the LV, the two parties compare their upcoming trajectory in a privacy preserving way. From this, both learn that they are going to share a section of their journeys. Thus, at time t_0, the candidate gets in line with the last FV and the ongoing co-presence verification process is started.

At time t_1, the CV has driven in line with the formation and continuously compared its past trajectory. This is used as proof for ongoing co-presence and since

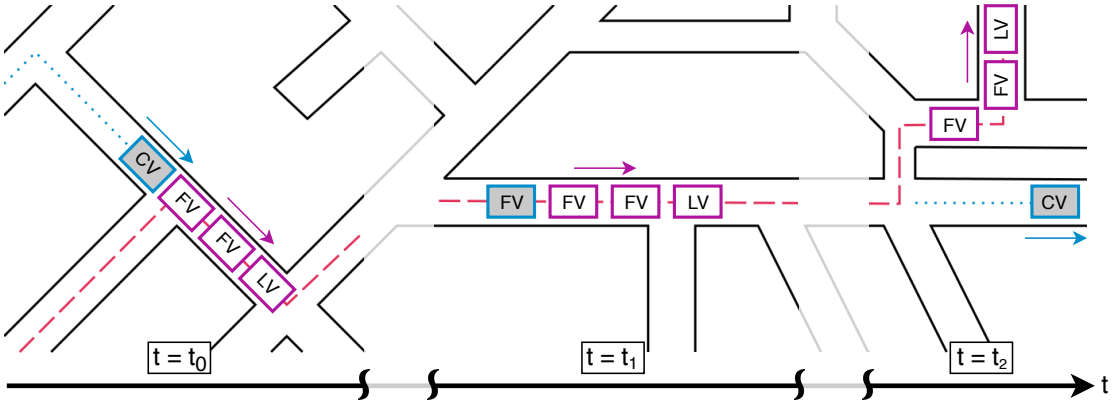


Figure 5.1: A vehicle formation with an LV, two FVs and a CV. At time t_0, the CV senses the group and decides to follow it. Next, at t_1 the LV and the CV shared the same trajectory which allowed the CV to become a following vehicle. Like the other FVs it now continuously verifies its co-presence using beacon messages that include its past trajectory. After t_2, a follower leaves the group, the lack of beacon messages between the LV and the FV terminates their sessions making it a CV.

the CV indeed followed the other vehicles, it gets promoted to a verified FV. Both vehicles continue exchanging beacon messages to mutually verify their co-presence.

As a member of the formation, a FV is granted access to their secure communication channel. It is used to confirm electronic signature information, i.e. their ECC public keys, allowing their OBUs to authenticate incoming VANET messages. This enables them to distinguish between messages from members of the formation and messages from stationary vehicles or vehicles driving in a different direction. Authenticating these messages is transparent to mechanisms like CACC and other safety applications, providing an additional layer of security. Further, we envision that the communication among FVs can be used to implement a more interwoven CACC which not only transfers acceleration and braking behaviour but may in the future even delegate steering to an autonomous LV. For convenience on longer journeys, member vehicles could also establish a secure communication channel among their passengers.

Eventually, FVs may reach their destination or divert from the formation's route and are hence no longer co-present. At time t_2, a deviation between a FV and the vehicle formation is shown. This is automatically detected by the LV who terminates

FV's session. The demoted vehicles is in future encounters again recognised as a CV.

5.3.2 System Model

In this section, we provide the assumptions on network participants and define the entities of our system: a vehicle formation, leader, member, candidate vehicles, and driving trajectories.

Network Entities

We assume that each RSU is in possession of a single ECC key pair and certificate, while OBUs hold a set of multiple credentials that can be used to sign messages. These ECDSA signatures allow all network entities to verify the authenticity and integrity of received messages. For privacy protection, a pseudonym scheme is deployed. Without imposing a limitation on the chosen pseudonym change strategy, we assume that a vehicle can arbitrarily switch to a new pseudonym. All network entities trust the credentials issuing RCA and can verify ECC credentials locally. Every vehicle holds a copy of a map for its geographic region which can for example be obtained from a publicly available source like the OpenStreetMap (OSM) project [157].

Vehicle Formation

The vehicle formation is a sequence of vehicles driving in succession. Each vehicle can have a different origin and destination, and therefore join and leave the formation at arbitrary times. However, at minimum vehicles share an intermediate section of their journey. To become a verified member, a vehicle is required to follow the group and provide evidence for this ongoing co-presence. Verified formation members are provided with a symmetric key that allows to establish a protected communication channel as well as sign messages using HMACs. These symmetric key based signatures allow a faster verification of the communication within the formation, e.g. CACC messages. Encrypted messages can be used to propagate pseudonym changes and hence speed up ECDSA verification of messages. In addition

to providing co-presence verification, this method also reduces false positives of Sybil attack detection systems by grouping different pseudonyms of the same vehicle.

Leader Vehicle

Any driver can opt-in to offer her vehicle as formation leader by announcing its location through periodically broadcast beacon messages. The privacy of these messages is inherently protected through the pseudonym scheme. When multiple potential LVs are within proximity, role assignment is determined by the provided location, where the first vehicle becomes the leader. This election could be complemented with a reputation system, based on driver behaviour or through commercial LV providers. A LV provides its FVs with the information to verify their mutual co-presence and is responsible for key management, i.e. issues key material to new FVs and terminates sessions as FVs leave the formation.

Following Vehicle

A FV is a vehicle that intends to share parts of its future route with the formation and has passed the ongoing co-presence verification. It has access to the internal communication using a symmetric key provided by the LV. The ongoing co-presence between a FV and its LV is continuously verified through beacon messages used for mutual trajectory comparison. FVs notify their LV when leaving the vehicle formation.

Candidate Vehicle

A CV is an unverified vehicle that is not associated with a leader vehicle. To join a vehicle formation, a CV compares its intended trajectory with the planned route of the LV. If their routes overlap, the CV starts the co-presence verification by continuously comparing its past trajectory with the LV. After the LV confirms their co-presence, the CV becomes a FV.

Trajectory

The intended trajectory of a vehicle is its future route through the road network, it can be derived from navigation information. This trajectory is used to probe whether two vehicles intend to travel along the same route. A vehicle's past trajectory is the route it took through the road network described by a sequence of turns and straight movement. As demonstrated in Chapter 3, these movement primitives can be acquired using IMU like gyroscopes. Depending on its length and the number of turns, a trajectory can uniquely describe a route on the map and implicitly a location. Ambiguities occur if multiple locations could be reached with the same trajectory. In our analysis, we call these collisions.

5.3.3 Adversary Model

For our threat model, we assume attackers who possess ECC credentials to produce valid VANET messages but are limited to polynomial time and thus cannot compromise asymmetric keys or symmetric cryptography. We recognise that an attacker can either pose as a CV or by disseminating beacon messages as an LV. In both cases, we assume a stationary attacker or an attacker who is passing by the victim in a different direction. The adversary is considered to be within wireless transmission range for the duration of the attack. The correct behaviour of computing devices is ensured using mechanisms such as remote attestation or trusted computing. For the evaluation of our approach, we categorise adversaries with respect to the following attack goals.

Identity and Location Privacy

This attacker aims to compromise the identity and future location privacy of a driver. Therefore, he uses the current location and heading of its victim to provide a valid trajectory for intended route comparison. He accomplishes this, if the forged trajectory matches the victim's trajectory. As a result, the attacker gets to know the future location of the victim. Additionally, the victim recognises him as currently co-located which triggers ongoing co-presence verification between her and the attacker.

Verified Following Vehicle

We consider two attacks on the co-presence verification mechanism. The attacker can either aim to become a FV without being co-present or try to remain a FV even though his vehicle already left the formation.

In the first scenario, the attacker has to find an intended trajectory that matches the victim's future route before being considered for ongoing co-presence verification. If successful or if he is already a verified FV, the attacker must provide a continuous stream of matching past trajectory segments. He can use the location where he encounters the victim in an attempt to infer its movement and derive this information.

An attacker who becomes a verified FV gains knowledge of formation members' identities allowing him to link their pseudonyms and thus defeat the pseudonym scheme. In case of a compromised or intentionally malicious FV, harmful information disseminated from within the vehicle formation can be mitigated using misbehaviour and intrusion detection; for this, we refer the reader to [18] as we consider it out of scope.

A FV could also relay past trajectory segments or disclose its cryptographic key to provide access to a third party. We do not consider such a scenario since the capabilities of a malicious FV which is physically present are the same as if a third party was involved, leaving the impact of such an attack unchanged.

5.3.4 Design Goals

In this section we set the design goals relevant to the implementation of the method described in our application scenario. The considerations related to security, privacy, and efficiency of our approach will be revisited in the security analysis and performance evaluation in Section 5.5.

Ongoing Co-presence Verification

Only vehicles that share the intended trajectory of the LV and are able to provide proof of ongoing co-presence must be recognised as a verified FV. An attacker who

supplies a fraudulent trajectory must be detected and rejected by the co-presence verification protocol. The system needs to ensure that the trajectory required to join the vehicle formation uniquely describes the current location, and is also resilient against inference by an attacker. Trajectory generation should not solely rely on external signals, for example GPS, as an attacker could spoof these signals [29] and make vehicles believe they are driving in formation.

The leader vehicle must establish a secure communication channel that is used to transmit the identities of FVs. It should not be possible for an attacker to read or insert messages. The credentials provided by the LV are tied to a vehicle's identity to fulfil non-repudiation and allow selective exclusion of formation members.

After a vehicle becomes a FV, they mutually and continuously verify their co-presence using trajectory beacon messages. The protocol design has to ensure that only the legitimate vehicles can provide this proof. In case a following vehicle leaves the formation, its credentials and identities have to be revoked from the set of trusted vehicles. Equally, if a leader vehicle abandons its group, all FVs have to invalidate their sessions. It is not sufficient for a member vehicle to indicate its departure from the formation as an attacker could suppress such messages and pretend to be co-present for longer than he actually is.

Identity and Location Privacy

The leakage of location information can reveal an individual's habits, activities, and even lead to physical harm [158]. Further, the home and work location of a driver can be inferred based on the route it has traversed [23, 24, 159]. This enables to reveal a driver's real identity through commuting habits [126], hence taking part in our protocol must not reveal a vehicle's intended or past trajectory. Parameters exchanged during a protocol run must not enable a new attack pathway allowing to compromise the victim's privacy through tracking or localisation. Thus, the protocol must work even without linkable VANET messages, in particular when beacon messages are periodically exchanged as this would allow an attacker to link the pseudonyms in use.

System Efficiency

Without human interaction, the system should only be triggered if two vehicles plan to share the next section of their journey. If they start the verification process, the number of messages required for its completion, and their computational complexity, has to be kept to a minimum. The duration a CV has to follow the formation until its co-presence has been verified should be minimised to maximise the benefits of becoming a FV. The overhead added by the ongoing co-presence verification through beacon messages should be negligible in comparison to the remaining VANET communication, e.g. CAMs.

5.4 Get-In-Line Architecture

In this section, we first describe the generation and comparison of intended and past trajectories, separately. Then, we present our co-presence verification protocol allowing us to perform these operations without compromising either of the parties' privacy. The protocol is designed to work independently from the participants' identities to enable the use of a pseudonym scheme while providing linkability of messages within the vehicle formation.

Our system consists of two phases: intended route matching and ongoing co-presence verification. The data flow between the different components is shown in Figure 5.2. During the first phase, two vehicles individually generate a representation of their intended trajectory (Section 5.4.2). After their successful privacy preserving comparison using our proposed protocol (Section 5.4.4), both move on to the second phase, where they capture their past trajectory (Section 5.4.3) which is subsequently compared, using the protocol from the first phase.

5.4.1 Trajectory Representation

We represent the trajectory of a vehicle as a sequence of movement segments, Seq. These segments are constructed as tuple that include a qualifier, p, and a quantifier, q. As shown in Equation 5.1, qualifiers distinguish between left (L) turns, right

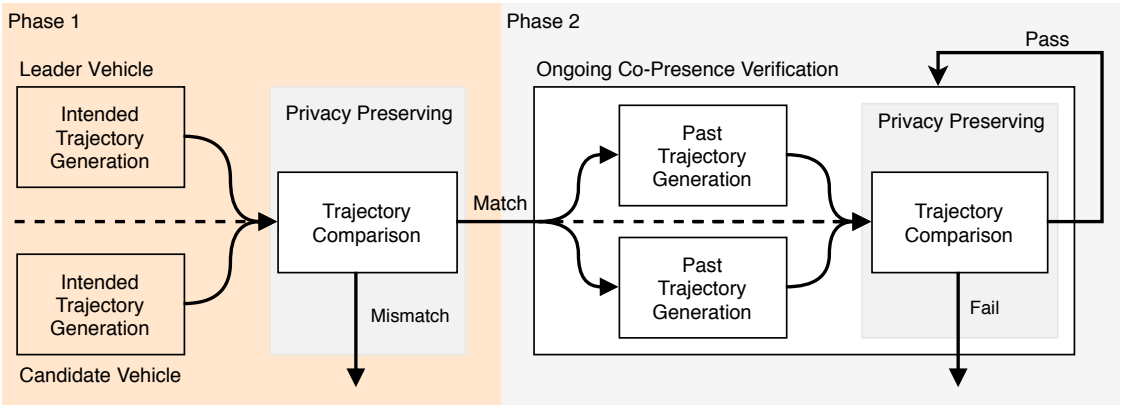


Figure 5.2: Map data and IMU measurements are used to generate intended and past trajectories. Their privacy preserving comparison allows two vehicles to first determine whether their future routes are going to overlap and to establish a symmetric key to secure further communication. After a CV has been verified and thus been promoted to a FV, their ongoing co-presence is continuously verified.

(R) turns, a inter-turn travel duration (D), as well as a inter-turn travel distance (I). We use the angle in degrees to quantify turns, duration in seconds for D, and distance in meters for I. Elements in the sequence are organised by their time of occurrence in chronological order with the most recent taking precedence, i.e. we define $T_{pa} = (p, q)_0$ as the newest segment in the sequence.

$$
\begin{aligned}
p &\in \{L, R, D, I\} \\
q &= \begin{cases} q^\circ \in [1, 180], & \text{if } p \in \{L, R\}. \\ q\ s \in \mathbb{N}, & \text{if } p \in \{D\}. \\ q\ m \in \mathbb{N}, & \text{if } p \in \{I\}. \end{cases} \\
Seq &= ((p, q)_0, (p, q)_1, (p, q)_2, ...) \\
T_{pa} &= (p, q)_0
\end{aligned}
\tag{5.1}
$$

5.4.2 Intended Trajectory

Given a map of the road network and a vehicle's current location, the intended trajectory is that section of its overall journey that has not yet been traversed. Assuming the driver is going to a specific destination, the journey can be determined using an integrated navigation system, for example based on OSM data. The comparison of two intended trajectories allows the participants to determine whether

their future routes overlap for at least the next d_{in} meters. As we will show in the evaluation section, an intended trajectory can be used to uniquely identify its starting location and thus assert the co-location of LV and CV when phase 2 is initiated.

Generation

Road networks are commonly represented as a graph, G, consisting of vertices (intersections and bends), V, and edges (roads), E. As shown in Equation 5.2, we derive the sequence Seq from a vehicle's intended trajectory T_{in}. Using the auxiliary functions $dist$ and $angle$, we compute inter-turn travel distances and turning angles along the vertices a vehicle passes over a given length d_{in} starting at v_0.

$$\begin{gathered} G = (V, E), V = (latitude, longitude) \\ E = (v_i, v_j) \mid v_i, v_j \in V, v_0 = start \\ T_{in} = (v_0, v_1, ..., v_n); \sum_{i=0}^{n-1} dist(v_i, v_{i+1}) \leq d_{in} \\ Seq = ((I, dist(v_0, v_1)), (L/R, angle(v_0, v_1, v_2)), (I, dist(v_1, v_2), ...) \end{gathered} \tag{5.2}$$

Figure 5.3 shows a road network based on OSM data. In this example, vehicles LV and CV are located on a straight segment leading to v_0. Both intend to take the same route highlighted with a red dashed line. To obtain the sequence representation Seq from T_{in}, the graph is traversed while turns taken and edges driven are recorded as L/R and I segments, respectively. For each vehicle, the traversal terminates, when it either reaches its destination or the cumulative length of the visited edges surpasses d_{in}.

Comparison

The map data underlying an intended trajectory is not subject to measurement errors. Hence the comparison of two instances $Seq_{LV} \stackrel{?}{=} Seq_{CV}$ can be computed as an exact match.



Figure 5.3: Map segment with an overlay of the road network. The LV and CV are currently driving towards v_0. Vertices and edges in red outline their future route used for intended trajectory comparison.

5.4.3 Past Trajectory

The past trajectory of a vehicle is used as proof for ongoing co-presence. Due to the inaccuracy of GPS coordinates [160], and their susceptibility to spoofing attacks [29], we use gyroscopes to identify a vehicle's turns and generate trajectory sequences. This also allows us to capture variations in driving speed and account for obstacles such as manoeuvring cars and buses, features difficult to perceive using a location source like GPS. Moreover, given that IMUs are on-board elements, they do not require an external signal source and readings can be processed directly on the device. During ongoing co-presence verification, the two participating vehicles periodically challenge each other to compare the latest segment of their past trajectory.

Generation

The basis for past trajectory generation is a sequence of turns associated with timestamps. Gyroscope data is captured from a vehicle's IMU with a sampling rate of 20 Hz [57] and high frequency noise is removed using a moving average filter. The trajectory features are then extracted in two stages.

For turn generation, we use angular velocity (rad/s) which is provided by a gyroscope along three orthogonal axis. Since a vehicle drives parallel to the ground, the axis pointing towards the sky is relevant to detect its turns. With only this axis as input, we use a peak detection algorithm [161] to identify spikes in the readings. We choose this approach over the continuous integration of the angular velocity used in Chapter 3 to mitigate cumulative errors inherent to inertial sensors due to thermal and mechanical disturbances [162]. As a minor disadvantage, this approach cannot recognise very slow turns. However, as we will show in our evaluation section, this does not appear to pose a problem when used for vehicles in an urban environment.

In the second stage, the resulting set of peaks is used to segment the time series and quantify each peak's turning angle through integration. An example output is shown in Figure 5.4 with green and red triangles, denoting right and left turns, and peak boundaries indicated with black lines. Orange areas provide the inter-turn travel duration which is obtained by computing the time elapsed between peak boundaries.

Comparison

To verify the ongoing co-presence of two vehicles, the latest segment of their past trajectories has to match. Their comparison is computed as an exact match $T_{pa}^{LV} \stackrel{?}{=} T_{pa}^{CV}$. However, due to variations in IMU measurements, this matching is affected by two types of errors: *order preserving* and *order disturbing*.

An order preserving error occurs when a past segment is of the correct quality but the quantities do not match. Such errors can be caused by sensor inaccuracies that lead to different turning angles, or by temporal variations like an obstacle on the road delaying one vehicle but not the other.

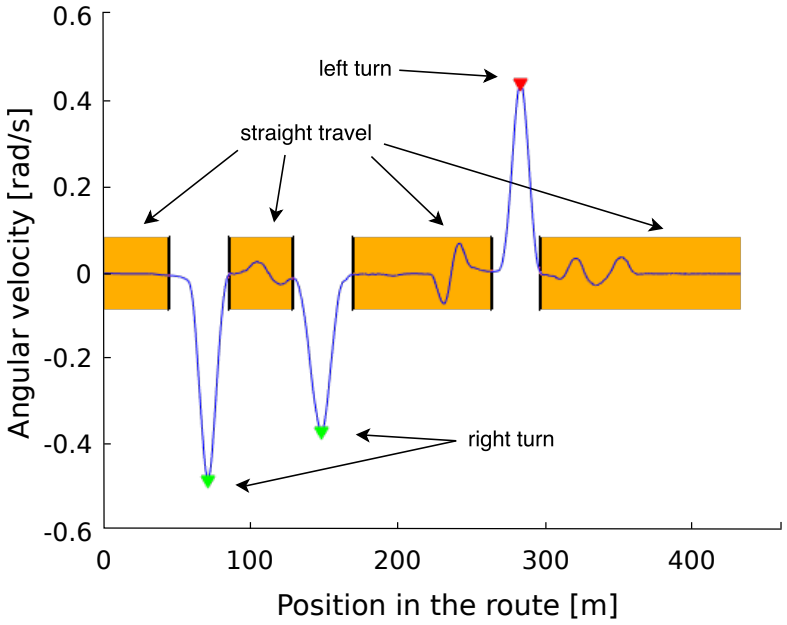


Figure 5.4: Gyroscope signal with peaks indicating turns where red denote left and green denote right turns. Orange areas were classified as inter-turn travel duration.

An order disturbing error changes a segment's position in the sequence and causes a mismatch for the qualifiers of the latest segments. The reason for such an error is a direction change that has not been recognised by the peak detection algorithm or an intermediate obstacle that needs to be circumvented.

We are going to investigate the impact of such errors on the number of falsely rejected verification attempts through experiments in our evaluation in Section 5.5.

5.4.4 Co-presence Verification Protocol

The protocol design implements the mutual authentication of the LV and the CV followed by the ongoing co-presence verification of FVs. Each operation is performed symmetrically and abort decisions can be made by both parties. To mitigate offline guessing attacks, the protocol requires the other party's involvement for each comparison attempt. The procedure itself is divided into two phases to accommodate the different stages of the CV: phase 1 matches intended trajectories of length d_{in} and establishes a symmetric key between the LV and CV. In phase 2, the initial co-presence of the two vehicles is verified with the past trajectory over the distance d_{pa}. If successful, the symmetric key is confirmed and the CV is promoted to

a FV after which ongoing co-presence is periodically verified. In case a FV leaves the transmission range of its LV or cannot provide the correct past trajectory, it gets de-authenticated and demoted to a CV. How values for d_{in} and d_{pa} can be determined and what impact they have on system performance is evaluated in Section 5.5.

We assume every vehicle has a set of pseudonyms, i.e. ECC public and private key pairs (K_v^+,K_v^-), where v denotes the associated identity. In the presentation of our protocol, we omit signature verification steps and assume that OBUs verify all messages using attached certificates and the VPKI.

Phase 1: Intended Trajectory Matching

The first phase is triggered when a CV receives the LV's beacon indicating the group's presence. Before ongoing co-presence verification (phase 2) is initiated, CV and LV determine whether they are going to share the next d_{in} meters of their journeys. Using the comparison procedure, both parties either established a symmetric key in case their future routes match or abort the protocol otherwise. This comparison is performed privately such that neither of the vehicles' future whereabouts are disclosed to the other. We provide a summary and description of the functions, notions, and parameters required for the protocol in Table 5.1. To start, LV and CV must have fresh RSA parameters (d, e, n) and a common understanding of the hash functions $H(\cdot)$, $H'(\cdot)$, and $H''(\cdot)$.

The different steps of this phase are outlined in Figure 5.5. Curly brackets denote transmitted messages and their signatures. In Step 1, LV and CV exchange the public parts of their RSA key pairs and agree on a d_{in}. In Step 2, both vehicles hash and blind their inputs to Y_{LV} and Y_{CV}. Upon receiving these elements in Step 3, each party signs them, to obtain Y'_{LV} and Y'_{CV}. The hashed and signed elements K_{LV} and K_{CV} are used to compute the tags T_{LV} and T_{CV} as well as to encrypt a *nonce* with a symmetric cipher, e.g. Advanced Encryption Standard (AES). In Step 4, both compare their respective tags $T \stackrel{?}{=} T'$ and, in case of a match, reverse the blinding operation to recover the keys K and use them to decrypt the nonces. If the tags do not match, the protocol is aborted. Finally, CV and LV derive the

Table 5.1: Pre-computed and public parameters of the private trajectory comparison protocol.

Parameter	Description
d_v	Private RSA exponent of vehicle $v \in \{LV, CV\}$
e_v	Public RSA exponent of vehicle $v \in \{LV, CV\}$
n_v	Public RSA modulus of vehicle $v \in \{LV, CV\}$
$AES(\cdot)_k, AES^{-1}(\cdot)_k$	Encrypt/decrypt data using AES under key k
$a \mid\mid b$	Concatenation of a and b
$H(\cdot)$	Full-domain hash function to project a trajectory segment into $\mathbb{Z}_n^\star$
$H'(\cdot), H''(\cdot)$	Cryptographic hash function $\{0,1\}^* \rightarrow \{0,1\}^\tau$
R_v	Generator $R \leftarrow \mathbb{Z}_n^\star$ used by vehicle $v \in \{LV, CV\}$

shared value $seed_0$ from $nonce_{LV}$ and $nonce_{CV}$. For a proof of the correctness of this protocol, we refer the reader to [163] who provide a detailed description how a similar construction can be used for private set intersection with data transfer.

Phase 2: Ongoing Co-presence Verification

Phase 2 of our protocol uses a continuous stream of beacons to verify the ongoing co-presence of a LV and another vehicle $v \in \{FV, CV\}$. These messages contain a vehicle's latest trajectory segment T_{pa} which is privately compared between the two. We deploy a slightly altered version of the protocol used in phase 1. By reusing the parameters transmitted in Step 1, we can omit that step. As for the comparison payload, we replace Seq with T_{pa}. The LV and v send mutual challenges whenever the qualifier of their T_{pa} changes.

Depending on the status of v, the outcome of the comparison is used differently. If v is a CV, it accumulates a sequence of shared values $seed_i$ and once all comparisons for a distance d_{pa} were successful, LV promotes the vehicle to a verified FV. Subsequently, both derive the vehicle specific symmetric key SK_v as shown in Equation 5.3.

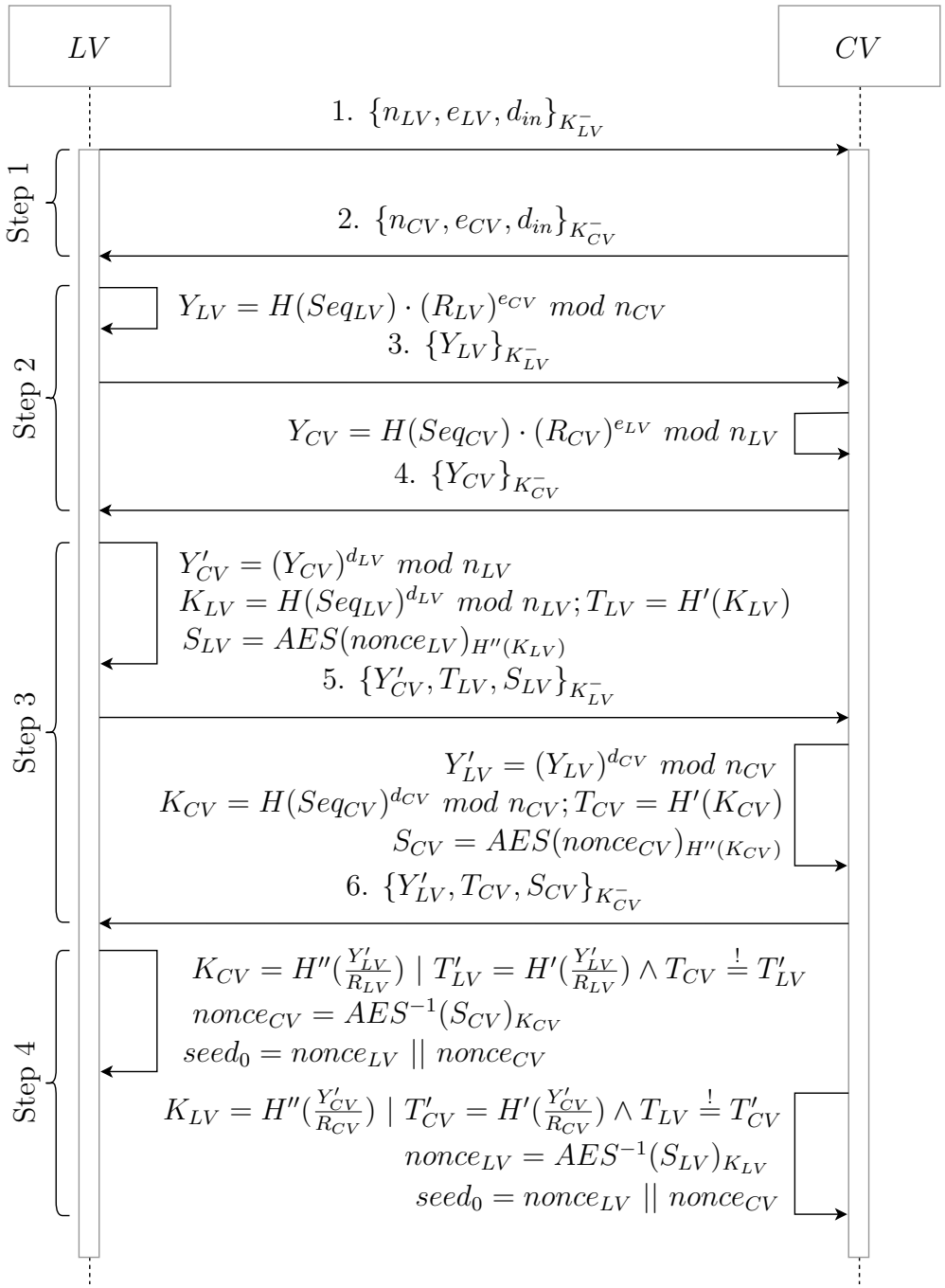


Figure 5.5: Phase 1 of the protocol privately compares the intended trajectories Seq_{CV} and Seq_{LV} for a given length d_{in}. Participants only learn about an exact match, which allows the derivation of the shared value, $seed_0$.

$$SK_v = H'(seed_0, ..., seed_i) \tag{5.3}$$

After completing this step, CV uses SK_v to obtain the symmetric group key

SK_G which allows its OBU to sign and encrypt messages intended for formation members using a HMAC and AES, respectively. The HMAC reduces message authentication significantly by not requiring the use of pseudonyms [114].

In case v is a FV, the same mechanisms is used to detect when v leaves the formation. From the protocol, the steps involving nonces can be omitted as both vehicles are only interested in the outcome of the tag comparison. The session is terminated if either the tags mismatch or the vehicles are out of each other's transmission range and no beacons are received. In both cases, a new group key SK_G is generated by the LV and distributed to the remaining FVs encrypted with their individual symmetric key.

The presented two-phase protocol first matches the intended and then the past trajectory of two vehicles in a privacy preserving way. After a CV has proven it followed the LV for d_{pa} meters, a shared vehicle specific symmetric key is established and CV gains access to the group key. Messages between protocol runs do not require to be explicitly attributed to a certain identity as the sequence of nonces implicitly ensures that only co-present vehicles receive the shared key. We analyse the security implications of d_{in} and d_{pa} in Section 5.5.2 and evaluate their impact on performance in Section 5.5.3.

5.5 Evaluation

The evaluation of our system is conducted in three steps: a map analysis, a security analysis, and real-world experiments. Using map data, we first show the generalisability of the approach across different geographic regions and quantify an attacker's search space for intended trajectories. Then, we use vehicle simulation as well as real-world experiments to analyse the resilience against attacks and the performance of our system. In each section, the respective results are presented and discussed.

5.5.1 Map Analysis

We use OpenStreetMap data [157] to study the topology of road networks across different cities. This allows us to investigate city specific characteristics and assess the generalisability of our approach. We are going to adapt the Manhattan mobility model which is subsequently used for trajectory generation and simulation. Next, the impact of the number of turns and overall length on trajectory uniqueness is evaluated. Finally, we estimate the search space complexity for an attacker who is unaware of the victim's location to show his likelihood to succeed in phase 1.

Road Network Topology

OpenStreetMap data provides a road network as a graph of vertices and edges. Vertices are annotated with GPS information and occur at intersections or bends in the road. Edges connect two vertices and specify a road type, for instance a cycling lane, walkway, or normal road. We only consider roads accessible by car and assume that roads can be traversed in both directions. In reality, one way streets and traffic circles may prohibit certain turns, however their number is negligible in comparison to the total number of roads in a city.

We characterise the layout of road networks by computing histograms for a total of four cities with large (Berlin, Singapore), medium (San Francisco), and small (Oxford) areal extent. These were chosen from North America, Europe, and Asia to eliminate country specific architectural differences. Figure 5.6 shows the segment considered for each city.

Using OSM information, the length of road segments and turning angles are computed as displayed in Figure 5.7. We define a turning angle as heading change, in this example a turn of 43° occurs when a vehicle turns around V_2 from V_1 to V_3.

Our results shown in Figure 5.8 are obtained through graph traversal and GPS based angle computation. They display the absolute number of turns for each angle between 0 and 180 degrees in all four cities.

In general, the distributions follow the same shape with a large number of small turns and a low number of large turns. A peak around the 90° mark can be observed



(a) Berlin - 1,200 km^2 **(b)** Singapore - 850 km^2

(c) San Francisco - 370 km^2 **(d)** Oxford - 180 km^2

Figure 5.6: We base our map analysis on the road networks of four cities: Berlin, Singapore, San Francisco, and Oxford. By choosing different geographical regions (North America, Europe, and Asia), construction epochs, and architectural styles we aim to analyse the effect of these road layouts on the security and performance of our system.

in all four cases which seems natural, particularly in North America due to the grid like road structure. However, the figures of the other cities suggest that the rectangular shape of buildings causes a similar effect in Europe and Asia. The larger number of smaller turns can be explained by straight sections and longer curves separated into several segments. These figures indicate that the turns in a city are largely independent of the city's location and areal size. As a result, they suggest that our approach is transferable between different cities.

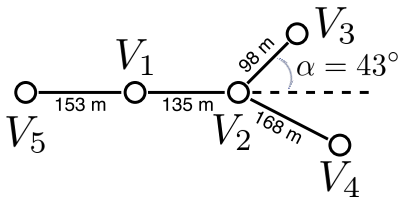


Figure 5.7: Example of a road segment with heading change α for a turn around V_2 from V_1 towards V_3. We use the Cosine-Haversine formula by Robusto [164] to compute the length of an edge using the GPS coordinates of its two vertices.

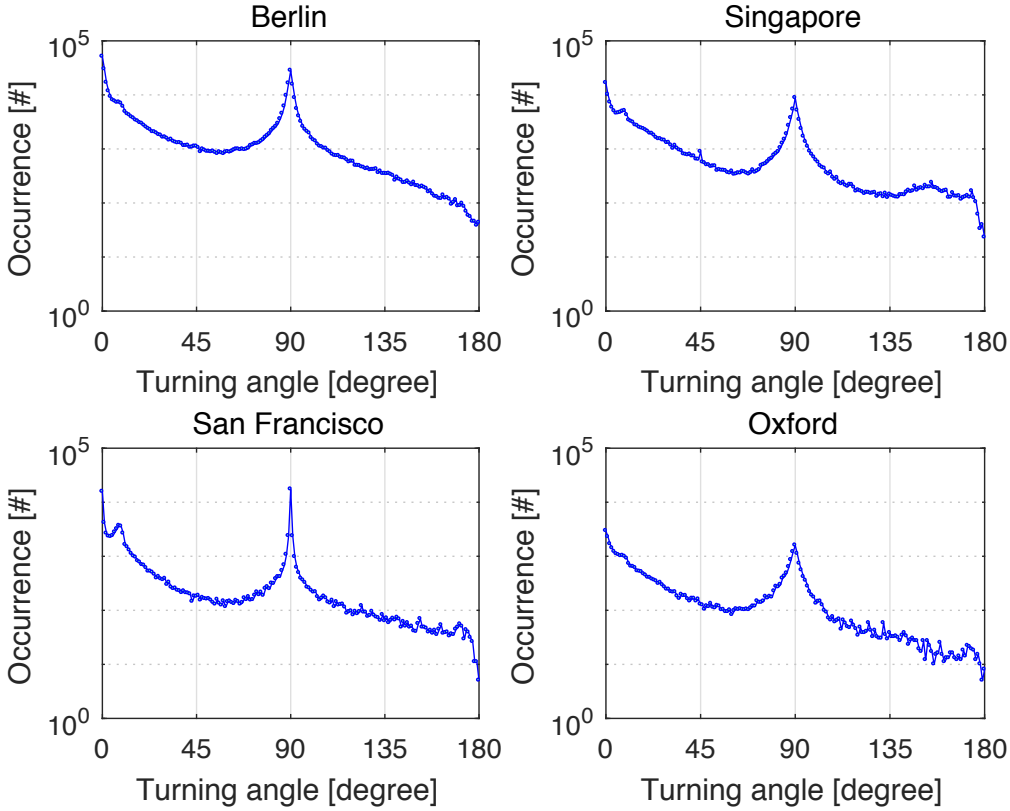


Figure 5.8: Turning angle distributions for the four cities. The absolute numbers of turns are given for each angle value from 0° to 180°.

Mobility Model

For the subsequent trajectory feature analysis and the simulation in the next section we adapt the Manhattan mobility model [165] for arbitrary turning angles as shown in Figure 5.9.

Straight movements are equally likely as taking a turn, and turns to the left are as likely as turns to the right. In the case, where a vehicle reaches an impasse, it is allowed to turn around. We consider angles ranging between +30° and −30° as

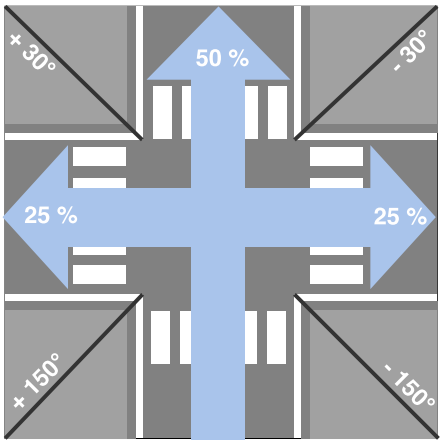


Figure 5.9: A vehicle moves straight with 50% probability, and turns left or right with 25% probability respectively. It is allowed to turn around if no other choice exists. Angles between 30° and 150° are turns while any smaller angles are forward and larger angles are considered backward movement.

straight movement, turns between $+30°$ and $+150°$, and $-30°$ and $-150°$ as left and right turns, respectively. Turns larger than 150° are considered backward movement.

Trajectory Feature Impact

To evaluate the expressiveness of turns and straight segments as driving trajectory features, we conduct two different experiments. For the first type, we keep the number of turns included in a trajectory constant, while for the second type, we keep the overall length of the trajectories constant. These sets are obtained by randomly choosing a vertex and simulating a vehicle driving in accordance with the mobility model. Afterwards, our algorithm performs a depth-first search for each vertex in the graph and enumerates the collisions of all trajectories of both sets.

We start by drawing 9000 trajectories with lengths of 50, 100, 200, 300, 400, 500, 1000, 1500, and 2000 meters for three groups that include 3, 6, and 9 turns. The number of collisions is averaged across the different lengths to obtain one data point for each of the three groups. The results are presented in Figure 5.10a, these indicate that with an increase in number of turns, the number of collisions decreases in all four cities. This suggests that combining multiple turns improves the uniqueness of a trajectory. Additionally, this parameter appears to be independent of the city size with San Francisco, the second to smallest, starting off at around

3000 collisions and Singapore, the largest, at around 1000 collisions. With an average of 60 collisions for trajectories with 9 turns, San Francisco is still above all other cities which exhibit less than 20 collisions at the same number of turns. Trajectories in Oxford that include 9 turns were found to have the least number of collisions. Given that Oxford has a smaller absolute number of turns, it is intuitive that its trajectories are more unique than when drawn from a larger city that offers more possibilities to find a similar turn sequences.

Next, the impact of a trajectory's overall length is evaluated by drawing 9000 trajectories of 9 different lengths from each road network. Figure 5.10b shows the results as average number of collisions per trajectory. The results suggest that this number depends on two factors: the area of the city and the length of the trajectory. With increasing areal size, a city has a larger number of roads which inherently allows for more collisions. However, if the length increases the exact combinations of turn and straight segments become rarer and the number of collisions decreases.

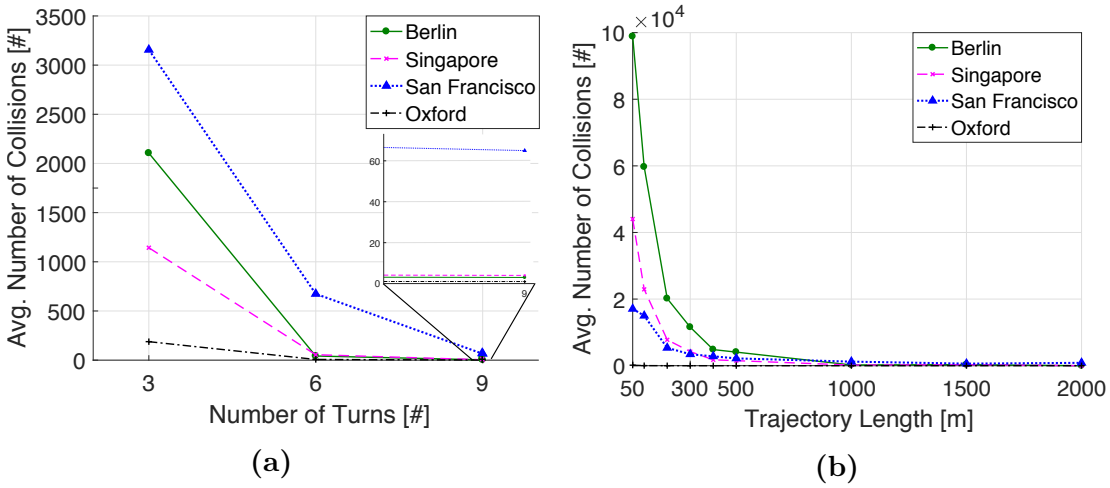


Figure 5.10: Trajectory feature impact: Number of turns included and overall length of a trajectory. (a) Average number of collisions per trajectory for each city from 9000 randomly chosen trajectories with three, six, and nine turns per trajectory. (b) Average number of collisions per trajectory for each city from 9000 randomly chosen trajectories of 9 different lengths.

Overall, these findings confirm that a trajectory through a road network can be described with turns and straight segments. Moreover, longer trajectories with more turns are more expressive.

Search Space Complexity

An attacker who is not aware of a CV's location can still attempt to pass phase 1 of our protocol by generating an intended trajectory from OSM data. The number of possible trajectories depends on the intended trajectory length parameter d_{in}. Based on the number of vertices and the average edge length in the considered road network, the search space size with respect to d_{in} can be computed as shown in Figure 5.11. The figures indicate that, without additional information, the number of possible trajectories increases with the size of the city. Thus, we find that for any trajectory longer than 20 meters even Oxford, the smallest of them, surpasses a hundred thousand possibilities. This makes such attacks unlikely to succeed even for small d_{in}. Hence, in the following security analysis we only consider attackers who are aware of their victim's location.

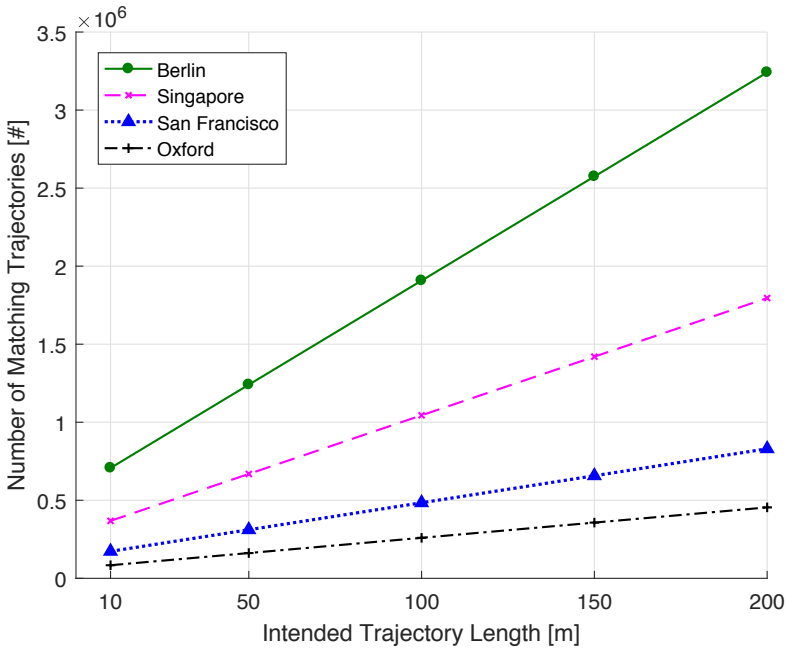


Figure 5.11: Search space size of intended trajectories for a location unaware attacker with respect to d_{in}. Average number of possibly matching trajectories in each city for $d_{in} \in [10, 50, 100, 150, 200]$ meters.

5.5.2 Security Analysis

We perform our security analysis in three stages. First, we investigate the effect of parameter d_{in} on the attack success rate against phase 1, i.e. an attacker who's

intended trajectory matches with the victim allowing him to proceed to phase 2. Next, to evaluate phase 2 of our protocol, we assess the de-authentication delay for a malicious FV who attempts to remain verified after leaving the vehicle formation. We implemented the approach of Studer *et al.* [20] and compare our results to show a significant improvement. We conclude with the cumulative FAR which reflects an attacker successfully passing both phases of our protocol. Each of these stages uses vehicle movement on the road networks of Berlin, Singapore, San Francisco, and Oxford based on aforementioned mobility model.

As indicated by the previous section, an attacker who is aware of the victim's location has a significantly higher likelihood to be successful. Therefore, we only consider malicious entities that are close to the victim, such as stationary vehicles or vehicles driving in the opposite direction. We note that we do not distinguish between a LV and the CV in the role of the victim in phase 1, since the attacks are analogue for both cases. Victims in phase 2 can either be the FV or an LV; a leader can abandon its vehicle formation and a following vehicle can depart from it, both cases have to be detected by the FVs or the LV, respectively.

Phase 1: Intended Trajectory Matching

The first phase of the protocol matches the destination of the LV and the CV for the d_{in} meters of their intended trajectory. The two vehicles privately compare this trajectory, and only commit to the next phase of the protocol in case of an exact match. An attacker uses the victim's current location, heading, and speed to generate a trajectory that is most likely to match the victim's intention. A successful attack, i.e. a false accept of an intended trajectory, compromises the future whereabouts of the victim and allows the attacker to proceed with an attack on phase 2 of the protocol.

Starting from a given vertex, the victim's intended trajectory is chosen based on our mobility model. Afterwards, 100 attackers are deployed at the same vertex which could be malicious vehicles or RSUs. Each attacker tries to infer the correct trajectory by traversing the graph and applying the probabilities from the mobility

model at each intersection. We simulate this attack for each vertex in the graph that has at least two adjacent vertices, i.e. it is not an impasse, and give the resulting FARs with respect to d_{in} in Figure 5.12. The curve for each city starts at the average edge length of the respective road network. The chance of correctly predicting the first intersection depends on the number of outgoing edges, i.e. how interconnected the roads are. For San Francisco, Singapore, and Berlin the figures suggest that there are predominantly T-junctions that lead to the attacker having close to 50% chance for successfully predicting T_{in} for small d_{in}. In general, the FAR drops for all four cities and reaches 5% for Oxford at 250 meters, Berlin, and San Francisco around 350 meters and Singapore at 450 meters. The slower decrease for Singapore suggests that its road network is less interconnected than for the other three cities and an attacker has less choices, i.e. is more likely to choose the correct trajectory, over the same distance.

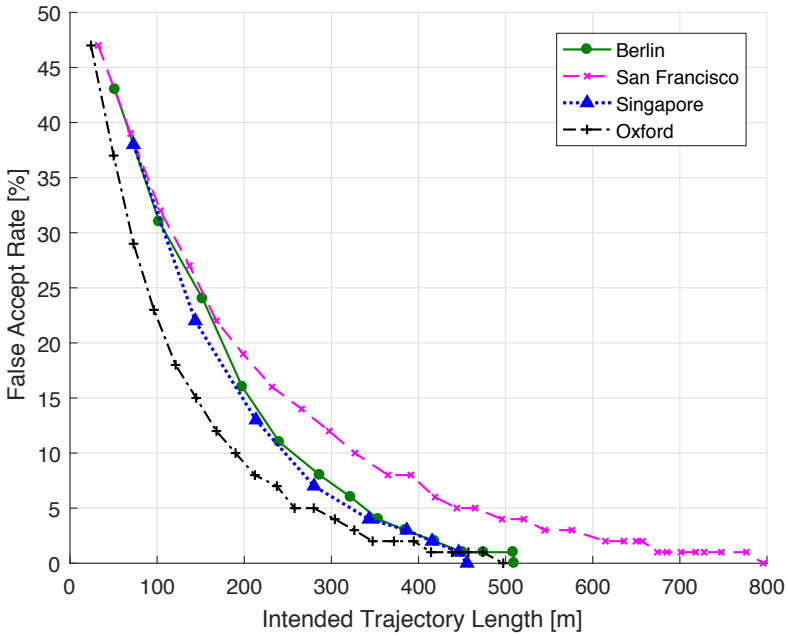


Figure 5.12: False accept rates in phase 1, intended trajectory matching, for all four cities when an attacker attempts to predict d_{in} meters of the victim's future route.

To summarise, by choosing a large enough value for d_{in}, only participants that actually intend to take the same route manage to proceed to the next phase of the protocol. For a small d_{in}, an attacker gets admitted to phase 2 more easily and obtains the next d_{in} meters of the victim's route.

Phase 2: Ongoing Co-presence Verification

In the second phase of our protocol, two vehicles that passed phase 1 verify their ongoing co-presence by periodically exchanging beacon messages to compare the latest segment of their past trajectory. A new CV has to successfully prove its ongoing co-presence for d_{pa} meters until it is promoted to a verified FV. The LV and all FVs continue this verification periodically. In case a trajectory segment does not match or one of the parties is unable to provide a beacon, e.g. is out of range, they are de-authenticated. A successful attack on this phase allows a malicious CV to become a verified FV, as well as a malicious FV to retain access to the group's secure communication channel. We assume that an attacker targeting this phase does not know the victim's past trajectory as he is otherwise by definition not violating the co-presence property. This leaves two types of adversaries: an attacker who passed phase 1 and is stationary or driving in a different direction within transmission range, and a malicious or misconfigured FV that does not notify the LV about its departure from the formation.

For the following analysis, we compare two approaches. One that is based on CAMs [20], and our system based on past trajectory beacons. Both assume a limited range of 250 meters for DSRC transmissions, a beaconing frequency of 1 Hz, and normal driving speeds within city limits. We model attacks as Bayesian Stackelberg games [166] with two participants: the victim and the adversary. The simulation is performed in turns. During each turn, the victim moves and then the attacker, both are governed by our mobility model. After a turn is completed, their trajectory segments are compared and the transmission range is checked. A false accept occurs if the attacker satisfies the range check and successfully matches the segment of the victim.

We start with a comparison of the two approaches in a simulation in Oxford to evaluate their de-authentication performance. The results in Figure 5.13a show the false accept rate with respect to the distance travelled by the victim. As can be seen, de-authentication occurs earlier for trajectory beacon approach since mismatches emerge while the transmission range requirement for status beacons still holds.

For both approaches, the FAR declines with increasing distance and falls below 5% after 290 meters for the past trajectory beacons and after 1500 meters for the status beacon based method. Since the past trajectory beacon method showed improved performance compared to status beacons in Oxford, we simulated this system on the remaining cities to validate our results. As displayed in Figure 5.13b, a similar behaviour is observed for all four cities. The different duration until the first de-authentication occurs can be explained by different average edge lengths across cities. Only after the first edge is traversed, the attacker has a chance to predict the wrong trajectory as we assume the adversary knows heading and location of its victim at the beginning of the attack. These results apply in the case of a malicious FV that does not notify the LV of its departure from the formation.

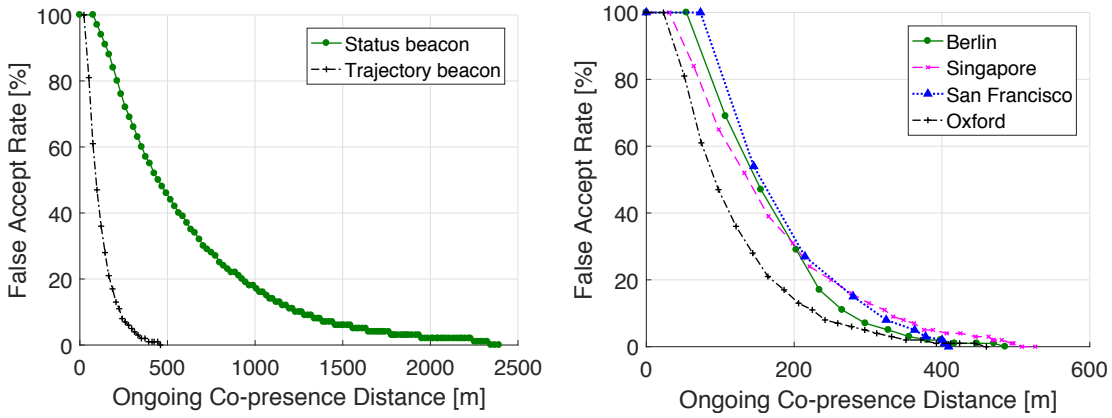


Figure 5.13: (a) Comparison of the two approaches, trajectory information significantly decrease the FAR for a given length allowing quicker verification of a CV and detection of a no longer co-present FV. (b) Trajectory beacon approach simulated in all four cities.

End-to-End Security

The probability of an attacker to gain verified co-presence with a given victim is computed by multiplying the FARs of the two phases. Since phase 2 can only begin after the vehicles drove for d_{in} meters, parameters d_{in} and d_{pa} are added to account for the overall distance travelled. The resulting combined FAR is shown in Figure 5.14. Oxford exhibits the fastest decline and reaches a FAR of 5% after 270 meters while Singapore and San Francisco fall below this mark at around 410

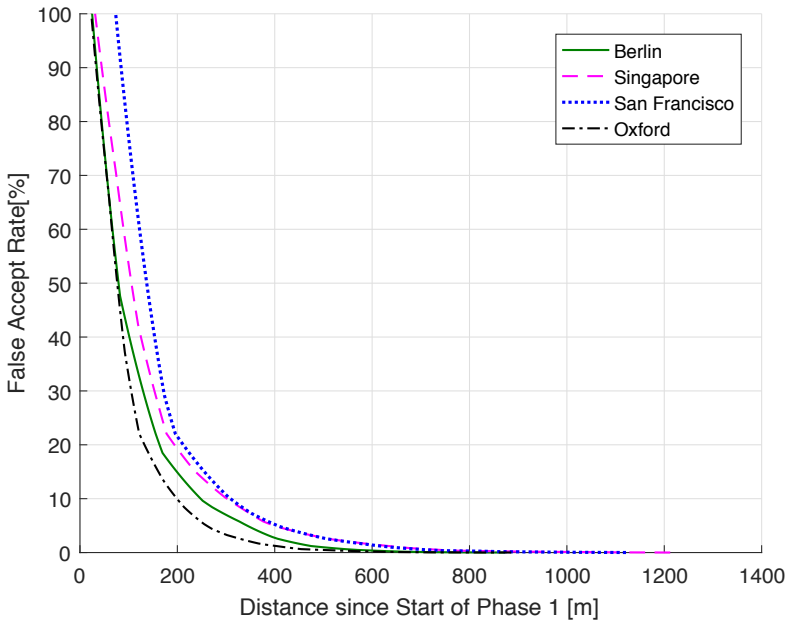


Figure 5.14: Combined false accept rates for phase 1 and 2 for all four cities with respect to distance since phase 1 initiation.

meters. In an urban environment, a vehicle takes around 20 and 30 seconds to travel such distances before a CV can be promoted to a FV. We note that the combined probability represents the likelihood of an attacker successfully passing through both phases of the protocol.

5.5.3 Experiments

The efficiency of phase 2 depends on the accuracy of the past trajectory generation algorithm. In this section, we present real-world experiments conducted to investigate the FRR, i.e. mismatches during the verification of two co-present vehicles. To acquire gyroscope readings, we implemented an Android application that transforms IMU measurements into trajectory sequences, T_{pa}. During each experiment run we obtained sequences which were subsequently compared, yielding the error rate.

As driving route, we selected the course outlined in Figure 5.15 with an overall length of 5.5 km. We rented the two cars shown in Figure 5.16a and placed smart phones in the centre console of each vehicle as displayed in Figure 5.16b. We repeated this experiment for a total of nine iterations throughout an entire day



Figure 5.15: Experimental route with a overall length of 5.5 km, selected for the analysis of phase 2. An impasse at A serves as start and endpoint.

featuring the morning rush hour as well as midday traffic conditions. An impasse at location A was used as start and endpoint to allow adjustments between runs.



(a) Experiment vehicles **(b)** Recording device

Figure 5.16: Experimental setup for the two vehicles scenario: (a) A 2016 Ford Focus (right) and a 2017 Vauxhall Astra (left) which were used as LV and CV (b) Smart phones were positioned in the centre console of each car.

Mismatches directly affect the performance of phase 2 since a vehicle that provides a segment that does not exactly match is de-authenticated. An example of gyroscope measurements for LV and CV from one of the experiment runs is given in Figure 5.17.

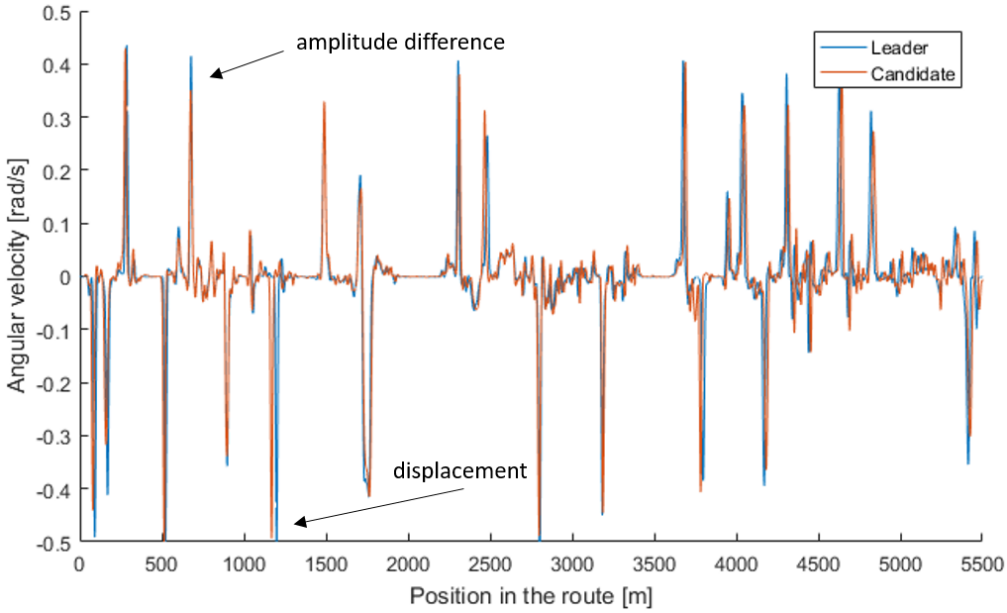


Figure 5.17: Gyroscope readings of LV and CV from an experiment run. Errors are introduced by differences in amplitude or displaced peaks.

As depicted, displacement or differences in amplitude can introduce errors in the past trajectory sequence, causing a false reject. The FRR for phase 2 depends on the required distance, d_{pa}, before a CV is promoted to a FV. Larger values have a smaller FAR, however they expose a higher chance to be affected by a mismatch while smaller d_{pa} are more robust and hence provide a lower FRR. This correlation between FAR and FRR is shown in Table 5.2.

Table 5.2: Summary of the experiment parameters and resulting error rates averaged across all runs.

Number of runs	9
Average run length	5256 m
FRR for 30% FAR	7%
FRR for 5% FAR	10%

We present the values for FARs of 30% ($d_{pa} = 120m$) and 5% ($d_{pa} = 250m$) which result in FRRs of 7% and 10%, respectively. The d_{pa} values are taken from the data for Oxford displayed in Figure 5.13b. As shown in the previous section, the combined FAR can be lowered depending on the value chosen for d_{in} and hence improve the overall FAR while still maintaining a low false reject rate.

Device Placement

We also tested whether the placement or the model and make of the used smart phone affects the gyroscope measurements. For this, we used a Nexus 6 and Google Pixel XL which were placed in the same vehicle at different positions. The Nexus 6 was attached as shown in Figure 5.16b while the Pixel XL was firmly attached to the floor of the trunk. Using this configuration, we collected data over 21.6 km within the city of Oxford, featuring 78 turns. The peak detection and angle extraction algorithms were run on the recorded signals and yielded a mean angle error of 0.028° and 0.027° while the maximum error was 0.057° and 0.069° for left and right turns, respectively (see Appendix Figure A.1). We conclude that the placement of the measuring device reflects the car's movement, as long as it is firmly attached.

5.6 Discussion

Additional Sequence Primitives

While we only focused on the use of low-cost inertial sensors to detect a vehicle's movement and turns, our system can easily be extended to incorporate additional trajectory features without changing the protocol. For example, the video feed of a dashboard camera could be used as input for feature extraction. Using an object detection algorithm [167], trajectories could be defined on a richer feature set, thus be more unique depending on the sights passed by a vehicle. Furthermore, the presence of specific wireless networks, Bluetooth beacons, or NFC tokens could also be modelled and used during the comparison procedure; if the vehicles are indeed taking the same path, such events should be detected by both of them.

Mobility Model

We used an adapted Manhattan mobility model to generate vehicle traces from OSM map information. Although this model has been widely used, the probabilities with which a driver chooses a road may not be static for all intersections. For example, the main road of a town is more likely to be followed by a driver than going through its residential area. A more realistic driving behaviour can be integrated

by assigning weight to the edges of the road graph during the simulation. These weights could be based on Google Maps traffic information or a microscopic traffic simulation package such as Simulation of Urban MObility (SUMO).

Inter-Formation Transitions

The presented protocol is based on the idea of vehicle platoons, however it does not support the transitions of FVs from one formation to another. Intuitively, if the co-presence was already verified by an LV, a FV's new formation leader should not have to run the full verification process again. While a key exchange between the two LVs is easily implemented, their trust relationship requires further research. A reputation system seems feasible, or the verification procedure needs to yield a record that can be verified by a third party without compromising the location history of the vehicles involved.

5.7 Summary

In this chapter, we presented our approach for ongoing co-presence verification in vehicle formations. Our two-phase protocol first allows vehicles to compare their future routes and then verifies their ongoing co-presence. The system performs both operations in a privacy preserving way and operates transparent to pseudonym schemes. After a candidate has successfully been promoted to a following vehicle, the group can authenticate VANET messages based on the trusted identities of formation members. This enables vehicles to defend against false location claims by an attacker who is stationary or driving in a different direction. Symmetric keys provided by a leader vehicle enable faster message verification and to establish a secure communication channel within a vehicle formation. Through real-world experiments and city-scale simulations, we showed the feasibility of our approach with acceptable false reject rates (10%) while maintaining low false accept rates (5%) after only 20-30 seconds of driving.

Although our protocol ensures that messages exchanged during the co-presence verification procedure do not have to be linkable, a pseudonym scheme was assumed

to actually provide passengers with location privacy. As stated in Chapter 4, however, pseudonym change strategies themselves are subject to attacks that aim to link a vehicle's identities using other, mandatory VANET messages (CAMs). In the next chapter, we shift the focus further towards the privacy aspect of VANETs. We are going to investigate the efficiency of cryptographic mix-zones, a safety-first pseudonym change strategy, its challenges under realistic traffic conditions and a potential remedy.

Hair loss is God's way of telling me I'm human.

— Bruce Willis

6

Cryptographic Mix-Zones for Private Pseudonym Change

Contents

6.1 System Proposal 114
6.2 Concept and Assumptions 116
6.2.1 Application Scenario: Chaff-based CMIX 116
6.2.2 System Model 118
6.2.3 Adversary Model 119
6.3 Chaff-based CMIX Scheme 120
6.3.1 Credential Acquisition 120
6.3.2 Chaff Notification 121
6.3.3 Mix-zone Participation 122
6.3.4 Cuckoo Filter Update 123
6.4 Security Analysis 124
6.4.1 Chaff Notifications 124
6.4.2 Mix-zone Interaction 124
6.4.3 Cuckoo Filter Update Propagation 125
6.5 Privacy Analysis and Evaluation 125
6.5.1 Exposure Metric 125
6.5.2 Quantitative Analysis 127
6.6 Performance Evaluation 130
6.7 Physical Layer Tracking 133
6.7.1 Methodology 133
6.7.2 Experimental Setup 133
6.7.3 Data Processing 135
6.7.4 Results 136
6.8 Discussion 136
6.9 Summary 137

In vehicular communication systems, CAMs provide contextual information required for transportation safety, e.g., EEBL, CCA, and LDW, and efficiency applications, such as CACC, presented in the previous chapter. However, as outlined in Section 4.6, without the appropriate design, these beacon messages introduce a new vector to compromise passenger privacy. The use of ephemeral credentials – *pseudonyms* – was therefore proposed, essentially to split a journey into unlinkable segments. Yet, this unlinkability is not easily ensured as an attacker can deploy techniques like syntactic and semantic linking (see Section 4.7). To protect segment transitions, two types of approaches, one based on silent and one based on encrypted periods (CMIXs) were proposed. We focus on the latter which is less invasive towards safety application. However, previous work in the area of CMIXs has only considered homogeneous traffic, disregarding variations in vehicle density due to differences in driver population, road layout, and time of day.

In this chapter, we investigate this open issue and present a novel scheme that works independent of vehicles' mobility patterns. In our evaluation, we compare our approach to an existing CMIX scheme under realistic traffic conditions in a large-scale simulation. Lastly, we provide an experimental investigation of physical layer tracking as yet another challenge faced by all cryptography based pseudonym change schemes. This chapter is based on our work in [168], [169], and [170].

6.1 System Proposal

Different solutions have been proposed to thwart pseudonym linking attacks. In silent period-based schemes, vehicles refrain from message transmission while changing pseudonyms [128, 136, 145]. However, the resulting lack of situational awareness diminishes the reliability of safety applications, such as collision avoidance [171], and greatly increases the probability of an accident [144]. Alternatively, CMIXs cover road segments in which messages are encrypted to conceal pseudonym changes and CAMs [146, 172]. Improved privacy protection was shown under idealised conditions,

i.e. homogeneous traffic and simplified road layouts. However, the resiliency of CMIX-based schemes against linking attacks highly depends on the driver population, vehicle arrival rates, and a uniform transition probability for vehicles traversing the mix-zones. More precisely, under realistic mobility patterns [173], real-world road network topologies [127], and mix-zone geometries [174], an attacker could infer enough information to harm user privacy [127, 146, 175]. Hence, these patterns have a significant impact on the efficiency and resiliency of CMIX schemes in real-world scenarios [168].

The system presented in this chapter addresses these problems by providing a mix-zone scheme with protection against syntactic and semantic linking attacks. In the past, chaff-based defences have been used to prevent traffic analysis in computer networks and provide anonymous communication [176]. Similarly, our system generates fictive *chaff* vehicles and adds their traces when needed, while it remains unobtrusive if sufficiently many real vehicles are present. This greatly improves privacy protection in situations with inherently low traffic density, e.g. in suburban areas, and during low traffic hours. Our protocols ensure that an external attacker cannot distinguish between real and chaff vehicles, while legitimate vehicles can recognise chaff message; this is important, because chaff vehicles (and messages) must not affect the operation of safety applications. The contributions of this chapter can be summarised as follows:

- We propose a scheme that introduces fictive *chaff* vehicles without interfering with the operation of existing safety applications. Our scheme smooths variations in road layout, time of day, and vehicle mobility patterns to improve user privacy. Unlike previous work, our construction considers the impact of a network of mix-zones on safety applications instead of one isolated mix-zone.

- We provide missing experimental results to show the impact of linking attacks by an eavesdropping adversary who uses features of the traffic flow. This completes the study by [140], which did not consider such attacks.

- In a quantitative analysis, we compare our chaff-based approach with an existing CMIX scheme to show the impact of variations in traffic density on passenger privacy. Our results under realistic traffic conditions show that by introducing fictive vehicles, we can enhance user privacy protection significantly.

- During an experimental investigation with two cars equipped with state-of-the-art OBUs from a European research project, we raise the issue of tracking using physical layer information of IEEE 802.11p transmissions. This implies new research questions regarding the efficacy of pseudonym change schemes.

6.2 Concept and Assumptions

In this section, an example use case is given to describe the system's application scenario. Afterwards, we state the assumptions regarding the involved VANET entities and the considered adversary model.

6.2.1 Application Scenario: Chaff-based CMIX

Our solution provides resilience to variations in traffic density, by introducing fictive *chaff vehicles*. Their purpose is to decrease the chances of an attacker compromising a victim's pseudonym change by linking its old to its new pseudo-identity. While chaff vehicles do not exist in reality, *chaff CAMs* are generated and signed with the private keys of *chaff pseudonyms* to materialise the presence of these vehicles. The basis for our scheme is formed by designated RSUs, positioned at locations where vehicles physically mix, e.g. at intersections [146], to create encrypted mix-zones for private pseudonym change. RSUs serve three purposes: maintaining a symmetric key to establish the encrypted region, generating messages resembling chaff vehicles, and ensuring that chaff pseudonyms do not interfere with safety applications.

Using the scenario in Figure 6.1, we describe the different tasks of our scheme. The four mix-zones (A, B, C, and D) indicated with dotted circles denote the

encrypted regions of the road network. For each mix-zone, an RSU manages chaff vehicle generation and key distribution for vehicles in proximity.

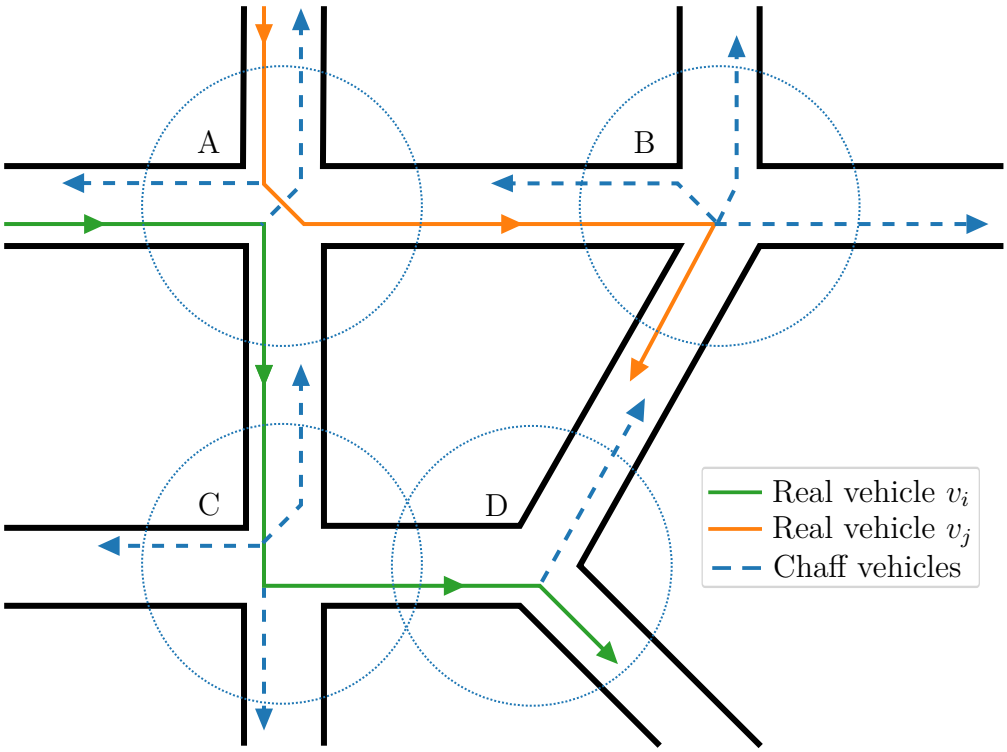


Figure 6.1: Network of four mix-zones: A, B, C, and D. Their encrypted regions are indicated with dotted blue lines. Green and orange solid lines show two real vehicles; each dashed blue line denotes the trace of a chaff vehicle.

After a vehicle entered a mix-zone, an RSU uses its knowledge about other vehicles in that zone, their intended trajectory, and the road layout to determine how many chaff vehicles are required. In the case of mix-zone A, two vehicles, indicated in orange and green, enter at the same time. Given these two *real* vehicles, their actual trajectories, and mix-zone exit points, two chaff vehicles need to be created to fill the remaining two exits (blue dashed lines); in fact, pretending that two 'real' vehicles exit from those exit points. The RSU responsible for mix-zone A uses location and heading from messages of previously observed real vehicles to generate synthetic CAMs, resembling the traces towards adjacent mix-zones. Message timestamps are adapted to emulate authentic location information and velocity, making chaff and real vehicles indistinguishable. While RSUs broadcast CAMs within their range directly, broadcasting chaff messages outside their coverage

needs to be delegated[1]. Neighbouring RSUs can undertake messages within their coverage while messages that are supposed to originate outside any RSU coverage can be delegated to real vehicles.

In order to preserve the correct functionality of safety applications, our scheme provides vehicles with information to identify chaff messages. Therefore, each RSU maintains a Cuckoo Filter (CF) [177] which it distributes to OBUs. Similar to Bloom Filters, CFs provide fast membership tests at the cost of an adjustable false positive rate, but in contrast support dynamic updates of the underlying set. We are going to use this property to keep the population of CFs constant to alleviate the introduced false positive rate. These CFs include the fingerprints of chaff pseudonyms used to sign chaff CAMs by adjacent mix-zone RSUs. When receiving a CAM, an OBU tests the attached pseudonym's fingerprint against its CF; if positive, the message is discarded. In our example scenario, the RSU, operating mix-zone B, maintains a CF that includes chaff pseudonyms used by the zones A, D and B itself.

For this to work, neighbouring RSUs and OBUs in transit between mix-zones have to be notified to update their CF. Due to propagation delays, these updates may inflict a delay until a chaff pseudonym can be used. However, to ensure effective privacy protection, an RSU cannot delay chaff message emission. As mitigation, RSUs can proactively include pseudonyms in CF updates to maintain a pool of these *ready-to-use* credentials for chaff message signature generation. We evaluate the size of such a pool and thus the size of CFs in Section 6.6.

6.2.2 System Model

We assume the setup as described in Section 4.5.4 for asymmetric message authentication based on a VPKI with LTCA and PCA. RSUs are authorised to request a set of chaff pseudonyms equivalent to the pseudonyms used by OBUs. The state-of-the-art VPKI deployments can easily be adapted to provide pseudonyms in a timely manner [178]. We further assume that all OBUs and RSUs are in possession of a Hardware Security Module (HSM). These modules serve as root of trust and

[1]We note that the encryption radius of a mix-zone is not equivalent to an RSU's transmission range. The effect of these radii and their impact on privacy is evaluated in Section 6.6.

allow the verification of the LTCA's and PCA's identities through the installed root certificate. Private keys stored in an HSM cannot be extracted and only one pseudonym per OBU can be active at a time. Before being sent, messages are signed using the OBU's private key corresponding to the currently valid pseudonym.

All RSUs are provided with a wired connection to communicate with VPKI entities and other RSUs. Each RSU is aware of its locality, the road layout, and its neighbouring RSUs. RSUs relay requests between OBUs and VPKI entities. Unless revoked, OBUs and RSUs are considered trusted.

6.2.3 Adversary Model

Adversarial models defined in the literature commonly differentiate between internal and external adversaries [132, 133]. The former include faulty, compromised, or malicious vehicles which have credentials and cryptographic material, making them a legitimate part of the system; they can deviate from system protocols. The external ones do not possess valid credentials, i.e. are unauthorised entities, but can interfere with the system's operation. From a different perspective, adversaries could be active, i.e. capable of injecting or modifying exchanged messages, or passive, i.e. capable of collecting information by eavesdropping the communication. In this chapter, we consider passive external adversaries. We assume an attacker who deploys off-the-shelf receivers along the road network. Performing OBU fingerprinting or angle-of-arrival computation requires significantly more sophisticated equipment. To give an outlook as to the feasibility of such attacks, we provide an experimental investigation in Section 6.7.

The attacker passively eavesdrops the communication and acquires information to compromise passenger privacy. Such information can be derived from CAMs, e.g., timing, velocity, heading, and location. Attackers could also collude and share separately collected information to compromise user privacy. Leveraging this information, they try to perform semantic linking by associating old and new pseudonym after a pseudonym transition. Especially because our scheme must not impair safety applications, a mechanism is provided for OBUs to distinguish

between fictive and real vehicle messages. This poses an attractive target for an attacker who aims to filter chaff messages out.

6.3 Chaff-based CMIX Scheme

The protocols required to implement our scheme are outlined in this section. Table 6.1 provides a summary and description of the functions and notions used.

Table 6.1: Protocol parameters summary and description.

C_{p^+}	Certificate corresponding to public key p^+
CF_e	Entity e's local Cuckoo Filter CF
$E_k(msg)$	Encryption of message msg using key k
$FP(\cdot)$	Fingerprint of a chaff pseudonym
LK_e^+, LK_e^-	Long-term public and private key of entity e
m	Number of vehicles in transit between two mix-zones
PK_e^+, PK_e^-	Pseudonymous public and private key pair of entity e
Pos_e	Position information (longitude, latitude) of entity e
R_r^{MIX}	Encryption radius of mix-zone r
Sig_{k^-}	Signature on the message using private key k^-
SK_r	Symmetric key valid for encryption inside mix-zone r
t_i	Fresh timestamp
Tr_v	Intended trajectory of vehicle v

6.3.1 Credential Acquisition

All OBUs and RSUs are assumed to have received long term credentials LK^+, LK^- after being registered at the LTCA. In order to obtain new pseudonyms, the LTCA provides a requesting entity with an authorising ticket which is sent to the PCA. Subsequently, the PCA responds with a set of pseudonyms. The LTCA only issues one ticket per OBU while RSUs can obtain multiple tickets. An RSU must not reuse a credential, otherwise an attacker would be able to classify it as a chaff pseudonym and discard the associated CAMs.

6.3.2 Chaff Notification

Chaff messages generated by an RSU r are limited to the road segments connecting it to adjacent mix-zones with RSUs o. Consequently, r's chaff pseudonyms can only be encountered along these segments. For example, vehicles in mix-zone B in Figure 6.1 can encounter chaff pseudonyms used by the mix-zones A and D. In order to not interfere with safety applications, these chaff pseudonyms are included in CFs provided to real vehicles in B's vicinity.

The protocol in Figure 6.2 is used by RSU r to notify its immediate neighbouring RSUs o by sending the fingerprints of pseudonyms $C_{PK^+_{chaff}}$ it intends to use (step 2.1, i.e. step 1 in Figure 2). The receiving RSUs add these fingerprints to their local CF_o which is going to be distributed to vehicles within their vicinity. Since vehicles in transit between any o and r might be temporarily without RSU coverage, this update is subject to delays. The RSUs o use m to indicate the number of vehicles that have left towards r since the last notification (step 2.2). If $m = 0$, no CFs need to be updated and the chaff pseudonym can be used directly. Otherwise, the chaff pseudonym cannot be used as these m vehicles would be unable to identify it as such, hence it is marked as pending. Once all relevant vehicles have received the CF update, PK^-_{chaff} is ready-to-use.

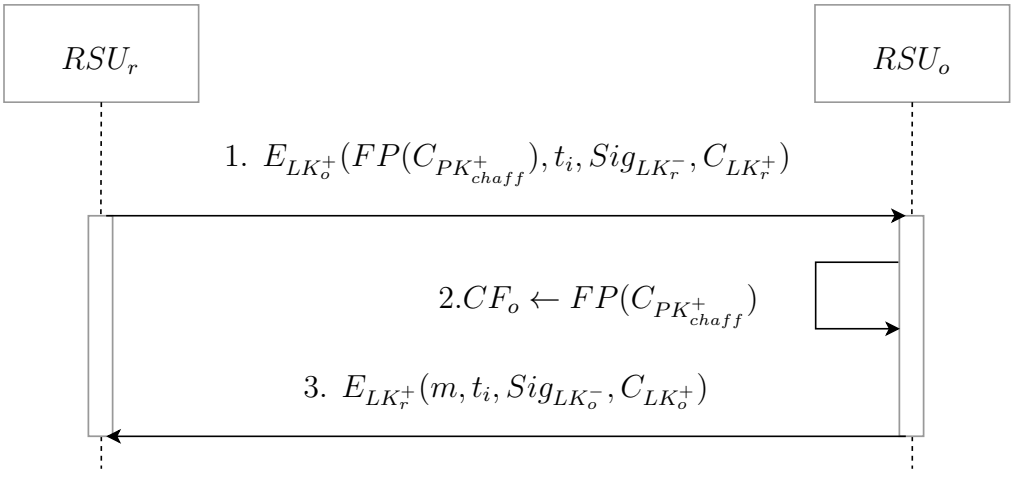


Figure 6.2: RSU r notifies its adjacent RSUs o about the fingerprint of a chaff pseudonym. The RSUs reply with m indicating how many vehicles are in transit towards r since the last notification.

When a chaff vehicle enters a mix-zone, the fingerprint of its chaff pseudonym is removed from the CF to reduce the CF's population. In our example, chaff pseudonyms used for chaff vehicles coming from B can be removed from CF_A once they enter A's encryption radius. When CFs are distributed to OBUs, a signature of the originating RSU and a timestamp is included.

6.3.3 Mix-zone Participation

The protocol needed for OBUs to interact with mix-zones, i.e. their RSUs, is provided in Figure 6.3. RSUs periodically broadcast the centre Pos_r and radius R_r^{MIX} of a mix-zone (step 3.1). When an OBU entered a mix-zone, it generates a new key pair (step 3.2) later used for CAM delegation. To join a mix-zone, the approaching vehicle provides its new public key $PK_v'^{+}$, its current cuckoo filter CF_v, and trajectory Tr_v through that road section (step 3.3). The RSU can mark pending pseudonyms included in CF_v as ready-to-use.

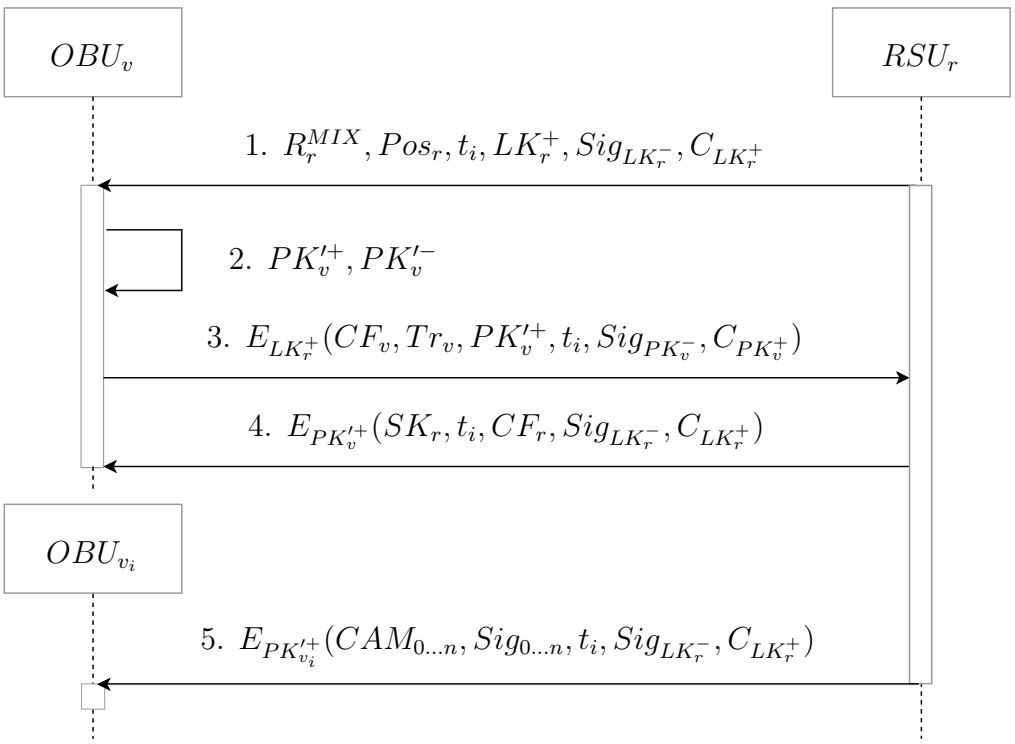


Figure 6.3: Mix-zones are advertised through periodic beacons. When approaching a mix-zone, OBUs prepare a new key pair, provide their intended trajectory, and obtain the symmetric mix-zone key.

Encrypted with the vehicle's public key and signed by the RSU, the mix-zone specific symmetric key and local CF_r is sent back (step 3.4). Based on Tr_v, the vehicle's speed and the other real vehicles in the mix-zone, the RSU generates the required chaff traces. The resulting CAMs are signed using ready-to-use chaff pseudonyms. Subsequently, RSU r uses its knowledge about the trajectory of all real vehicles v_i in the mix-zone to delegate chaff CAMs that are required to originate outside its coverage. In step 3.5, these CAMs and signatures are delegated to v_i by sending them encrypted with v_i's public key $PK'^{+}_{v_i}$. The OBU of v_i is going to broadcast these CAMs based on message timestamps.

6.3.4 Cuckoo Filter Update

In order to reduce the duration until a pseudonym is ready-to-use, we use epidemic propagation to disseminate CF updates between vehicles (see Figure 6.4). When a vehicle v_i encounters a new peer v_j, it initiates the forwarding protocol by sending CF_r (step 4.1). After OBU v_j responds with its $CF_{r'}$ from the same mix-zone (step 4.2), both OBUs compare the timestamps of the CF they received. Once the signature attached to the CFs is verified, v_i and v_j update their CFs to the more recent one. Since vehicles located outside the range of a given mix-zone's RSU r are unaware of SK_r, we piggyback SK_r on this exchange.

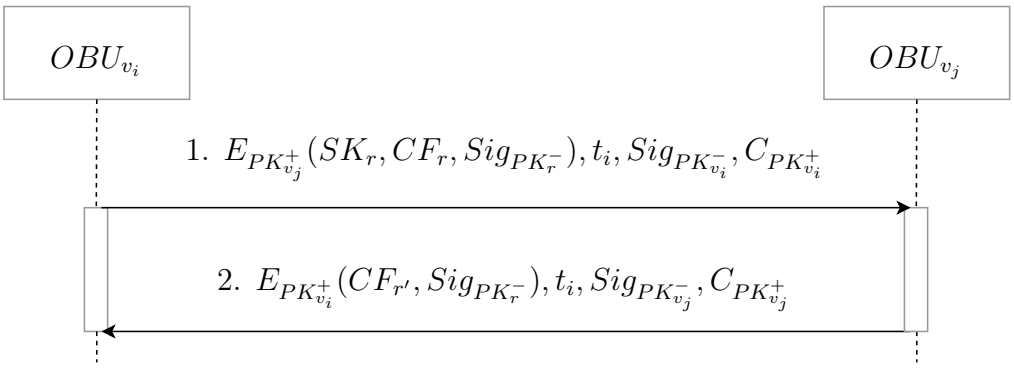


Figure 6.4: OBUs forward CF updates and mix-zone keys epidemically outside RSU coverage.

6.4 Security Analysis

Throughout this analysis, we assume that messages are digitally signed under the private key of the currently valid pseudonym of an OBU or the LTC of the RSUs. They also include timestamps, thus protocol interactions are protected against injection and replay attacks.

6.4.1 Chaff Notifications

RSUs provide fingerprints of chaff pseudonyms to adjacent mix-zones to ensure safety application functionality. In the hands of an attacker, these fingerprints allow him to discard chaff messages and degrade the privacy protection offered by our scheme. Equally, if an attacker manages to inject fingerprints of real pseudonyms, OBUs would ignore actually relevant messages, jeopardising passenger safety. Hence, two RSUs exchange chaff notifications encrypted under the receiver's public key and provide a signature for integrity. The number of real vehicles in transit between two mix-zones m is also to be encrypted, otherwise it allows an attacker to reason about the number of chaff vehicles. For example, given $m = 0$, an attacker would conclude that no real vehicles are present and all observed messages originate from chaff vehicles.

6.4.2 Mix-zone Interaction

Four types of information are transmitted when a vehicle interacts with a mix-zone's RSU: (1) the vehicle's new public key, (2) its intended route, (3) the vehicle's and RSU's CFs, and (4) chaff CAM delegations. If in possession of (1) or (2), it is trivial for an attacker to link old and new pseudonym of a vehicle. (3) and (4) enable the attacker to identify chaff pseudonyms and discard chaff CAMs. An OBU encrypts (1), (2), and (3) with the RSU's public key during protocol initiation, thus protecting them. The response providing the mix-zone's symmetric key and the RSU's CF is encrypted with the vehicle's public key. To guard CAM delegation (4), the RSU uses the target vehicle's public key to encrypt the communication.

6.4.3 Cuckoo Filter Update Propagation

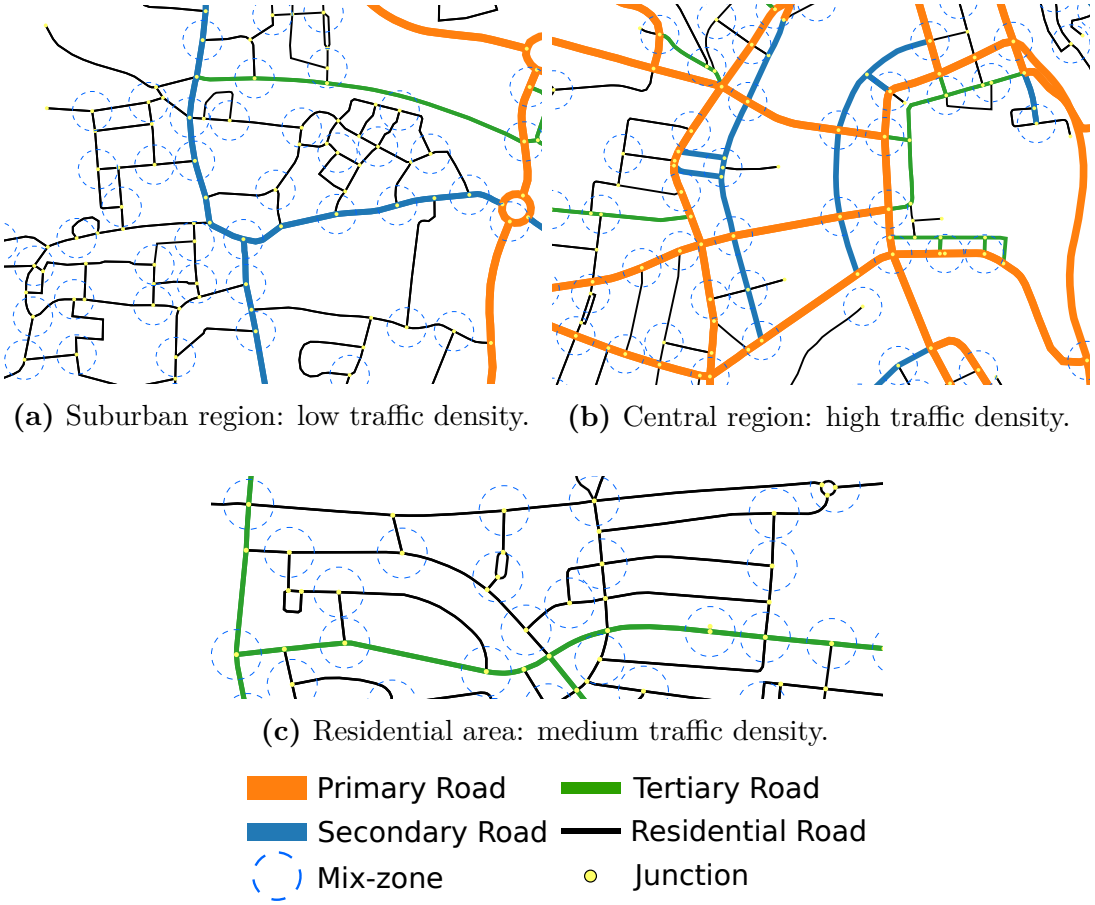
Our scheme epidemically propagates CF updates and mix-zone keys. A CF obtained by an attacker would enable the exclusion of all messages signed with the included chaff pseudonyms and hence defeat our scheme. A leaked mix-zone key would turn an external adversary into an internal one, effectively disabling that zone. To prevent an eavesdropper from acquiring either of these information, OBUs encrypt their communication using the respective receiver's public key. The injection of forged CFs is mitigated by attaching a signature of the originating RSU. Since CFs have a fixed size (see Section 6.6) and are encrypted, an attacker cannot infer the number of active chaff vehicles from them.

6.5 Privacy Analysis and Evaluation

Our evaluation assesses the privacy protection achieved under three scenarios with different traffic densities (see Figure 6.5). These were chosen due to their range in road types and differences in driver population. We compare our chaff-based approach to the original CMIX [146] scheme that did not consider sparse vehicle availability, hereafter termed the *baseline* scheme.

6.5.1 Exposure Metric

In previous work, metrics capturing a vehicle's privacy, such as anonymity set size and tracking entropy, were used [143, 146, 149]. These assume vehicles are dispersed according to a Gaussian distribution and that mix-zone transition probabilities are normally distributed, which limits their expressiveness under real traffic conditions [179]. For a more realistic privacy estimate, we define an exposure metric based on a vehicle's route length and the number of mix-zones traversed during a trip. It captures the impact of a linked change from pseudonym p to a new pseudonym p' by weighing it with the distance the vehicle will remain using p'. This puts the privacy compromise in relation with the overall trip, reflecting the impact of attacks as shown by [141] more accurately. For example, if the vehicle changes from



(a) Suburban region: low traffic density. **(b)** Central region: high traffic density.

(c) Residential area: medium traffic density.

Figure 6.5: The areas used for the simulation expose individual mobility patterns due to their driver population and road layout. Road capacities are colour coded, yellow points show junctions while dashed circles indicate encrypted mix-zones with $R^{MIX} = 50$m.

p to another pseudonym p'' after a short period, the impact of a linking p to p' is smaller than if p' was used for a longer period as a larger piece of the journey can be attributed to the same vehicle. In Table 6.2, we provide the notation and parameters of a road network which we model as a graph of nodes (junctions) N and edges (roads) E. A subset M of nodes denotes the mix-zones of the network. Equation 6.1 shows how the privacy degradation $P(T_v)$ for vehicle v during a trip T is computed.

$$P(T_v) = \frac{\sum_{i=0}^{m} l(e_i) * p_v(n_i)}{|T_v|} \tag{6.1}$$

When traversing node n_i, the length of the exiting edge $l(e_i)$ is weighted by the attacker's probability p_v of successfully linking v's pseudonyms across the

Table 6.2: Metric parameters summary and description.

E	Edges $E = (n_i, n_j); n_{i,j} \in N$
$l(e)$	Length of $e \in E$
N	Junctions of the road network
M	Mix-zones $M \subseteq N$
$p_v(n)$	Probability to link v's pseudonyms when traversing $n \in M$
T_v	Trip of vehicle v, $T_v = (n_0, ..., n_i); n_i \in N$
$\|T\|$	Total length of trip T

transition. In case $n_i \notin M$, this probability is set to 1.0, otherwise it is computed depending on the applied scheme. For the baseline scheme, traffic flow features are used, such as transition duration, transition probability, and the number of vehicles in the mix-zone [146]. For our chaff-based scheme, the number of vehicles and the generated chaff messages form the basis of the probability computation. Finally, the total length of a trip $|T_v|$ is used to normalise the exposure for vehicles with different trip lengths.

6.5.2 Quantitative Analysis

For our simulations, we extended the implementation of the PREXT[2] project [140] based on the Omnet++[3] and the Veins framework[4]. We used the SUMO traffic simulator to produce realistic mobility traces according to the Luxembourg SUMO Traffic (LuST) dataset [173] for vehicle trajectories in the city of Luxembourg. The characterising features of the three selected regions are provided in Table 6.3.

Table 6.3: Key figures for the simulated regions ($R^{MIX} = 50$ m)

Parameter	**Suburban**	**Residential**	**Central**
Area	1.31 km^2	0.61 km^2	1.39 km^2
Junctions	69	34	61
Mix-zones	31	18	28
Average number of vehicles per mix-zone	1825	4631	6500

[2]https://github.com/karim-emara/PREXT
[3]https://omnetpp.org/
[4]http://veins.car2x.org/

We show the area, the number of junctions, the number of mix-zones, and the average traffic density. Since an optimal mix-zone placement is NP-complete [180], we used a heuristic-based approach which maximises the number of mix-zones across a map for a given encryption radius. For our initial comparison, we chose $R^{MIX} = 50$ meters. The resulting mix-zones for each region are highlighted with dashed circles in Figure 6.5. Mix-zones are placed such that their encrypted areas do not overlap, otherwise, they simply merge into one larger area. The effect of different values for R^{MIX} will be evaluated in the second half of this section. We compared three different attack strategies, choosing mix-zones randomly, by traffic density, and by the number of roads exiting a mix-zone. Since selecting mix-zones by traffic density yielded the best results for the attacker, we evaluated assuming this strategy. After simulating the baseline and our scheme for the three scenarios over a full-day period, we obtained the results depicted in Figure 6.6. The figure provides the exposure values for the 90$^{\text{th}}$ percentile of the vehicle population.

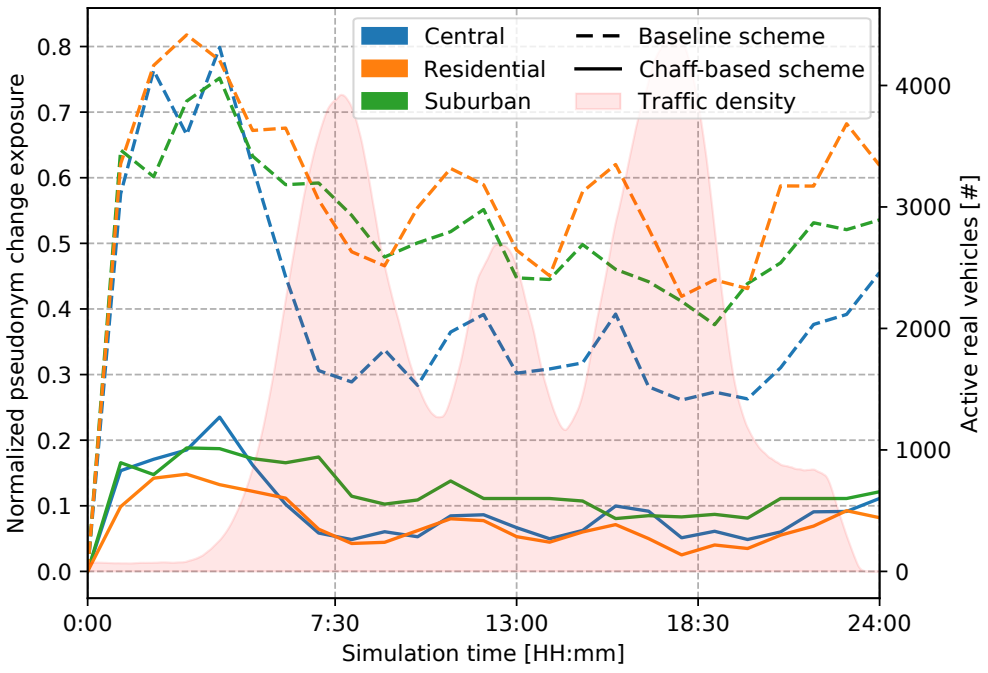


Figure 6.6: Privacy degradation comparison: lines show exposure values for the 90$^{\text{th}}$ percentile of the vehicle population.

We found that the privacy exposure is inversely correlated with real vehicle density. This can be seen during the rush hours around 7:30, 13:00, and 18:30

which coincide with a reduction in exposure values. Due to very low traffic density from midnight to 5:00 am, this period provides vehicles the least privacy protection. Regional variations, for instance between the central and residential area, also contribute to differences in privacy exposure levels. Despite a slightly elevated privacy compromise in the morning hours, our approach appears more resilient against these fluctuations providing a homogeneous and overall higher level of privacy protection.

Next, we evaluated the effect of different encryption radii on the performance of both schemes. We considered an attacker who has several observation spots with limited coverage [129] and the worst case, a global attacker[5] in the suburban and central region. As shown in Figure 6.7, we conducted simulations for $R^{MIX} = [50, 100, 150, 200, 250]$ meters.

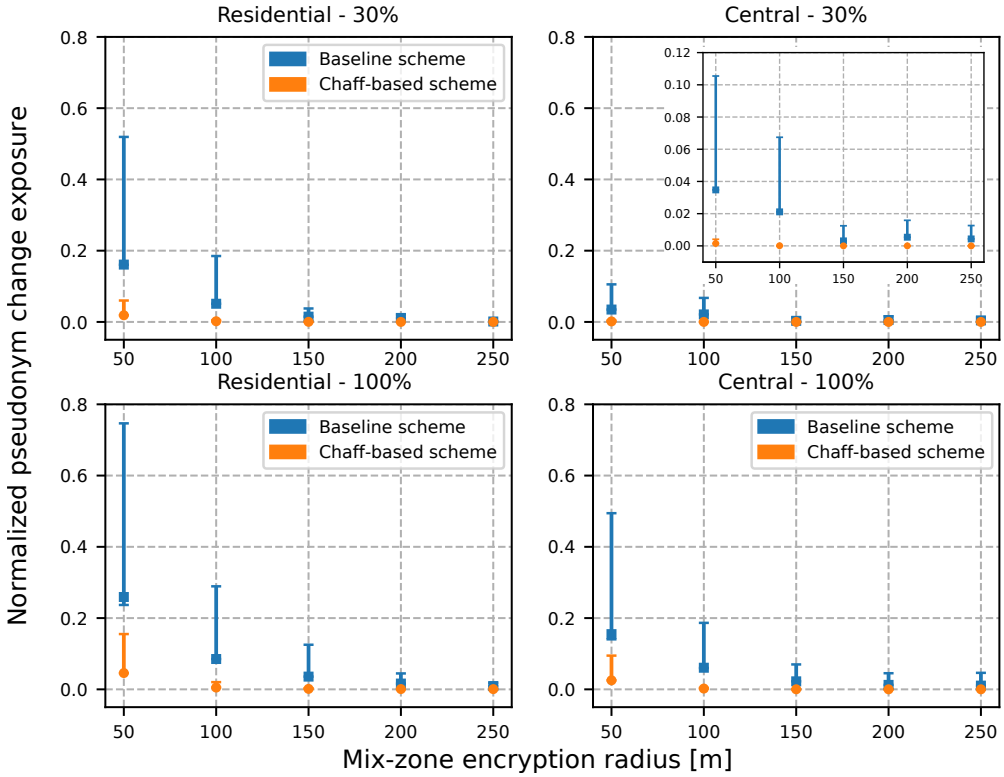


Figure 6.7: Effect of mix-zone encryption range on vehicle exposure. The attackers observe 30% and 100% of the mix-zones.

[5]As mentioned in Chapter 4, the VANET communication standard was designed to be open and close to the WiFi frequency spectrum for easy adoption, making this attacker conceivable.

We considered an attacker who has several observation spots with limited coverage [129] and a global attacker, capable of covering the entire network, in the residential and central region. These regions were chosen since they exhibited the best and worst privacy protection in our analysis with $R^{MIX} = 50$. We found that user privacy improves with increasing encryption range; however, the baseline scheme converges slower than our chaff-based scheme and exposes larger variations, confirming its dependence on mobility patterns. After an initial improvement in privacy protection by increasing R^{MIX} from 50 to 100 meters for the chaff-based scheme, the privacy protection remains optimal thereafter. These figures indicate that while the baseline scheme offers more user privacy with a larger R^{MIX}, a chaff-based scheme can operate almost independent of the value chosen for R^{MIX}. A smaller R^{MIX} means that a vehicle enters and leaves more mix-zones, consequently its journey is divided into more segments which offers more privacy protection according to Freudiger *et al.* [146]. Additionally, if internal attackers are considered, the presence of more mix-zones makes it more difficult for an attacker to collect all mix-zone specific keys.

6.6 Performance Evaluation

We evaluate the performance of our system using three metrics: chaff pseudonym pool size, the number of simultaneously active chaff pseudonyms, and the RSU signature generation overhead. Considering different values for R^{MIX}, the remainder of this section discusses resource requirements on the basis of these metrics. The results acquired over the whole 24h simulation are summarised in Table 6.4.

Table 6.4: Different encryption radii affect the number of simultaneously active chaff pseudonyms, chaff message generation, and pseudonym pool size.

Mix-zone encryption radius [m]	**50**	**100**	**150**	**200**	**250**
Max. Active chaff pseudonyms [#]	68	176	165	99	66
Max. CAM generation $[msg/s]$	240	848	1321	831	634
Max. Pseudonym pool size [#]	5	4	3	4	3

For the number of active chaff vehicles, we are interested in two aspects. Firstly, whether real vehicles can alleviate the number of required chaff vehicles, and secondly the absolute computation overhead experienced by an RSU for chaff CAM signature generation. Given the influx of vehicles into a mix-zone under consideration of the availability of real vehicles, we can compute the number of required chaff vehicles. Intuitively, this number should be inversely correlated with the traffic density. Figure 6.8 shows each mix-zone's active chaff pseudonyms per real vehicle.

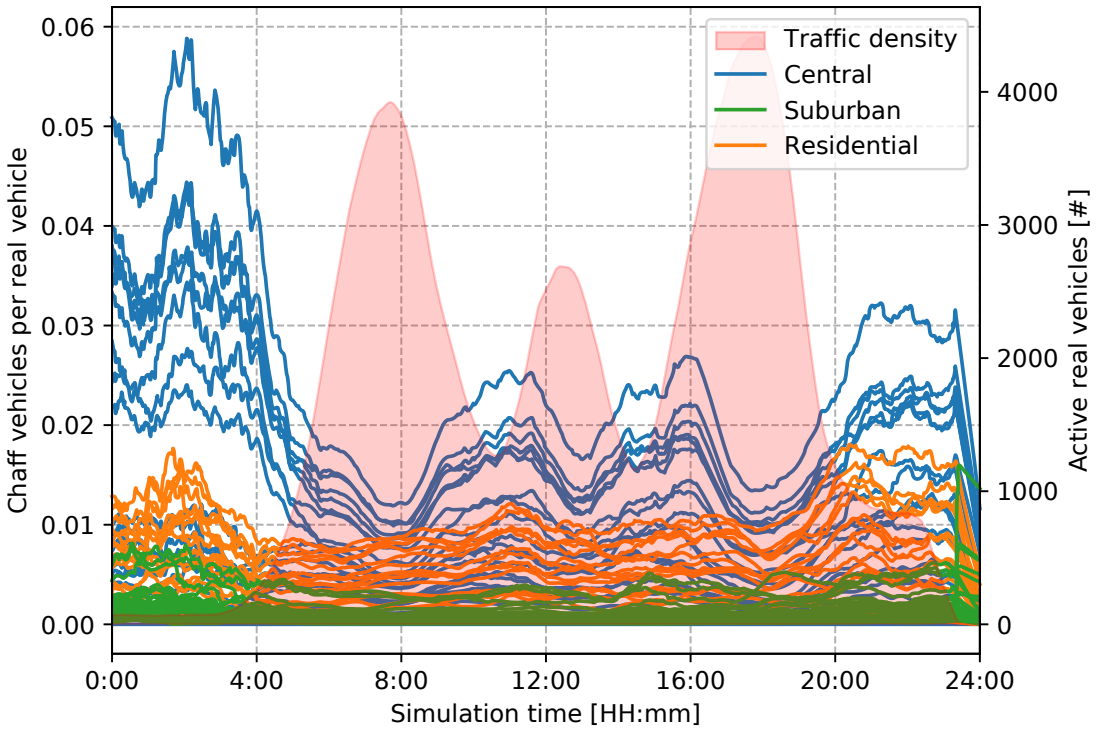


Figure 6.8: Active chaff vehicles per real vehicle. The shaded area denotes number of real vehicles. Each line represents one mix-zone: 28, 31, 18 for the central, suburban, and residential region.

As expected, we found that with an increase in real vehicle density during peak hours (around 7:30, 13:00 and 17:30), less chaff vehicles per real vehicle are required to achieve the same level of privacy protection. Secondly, based on trace length and the number of required chaff vehicles, we can compute the rate with which RSUs have to sign chaff messages. Studies on Nexcom boxes (Dual-core 1.66 GHz, 1 GB memory) from the PRESERVE project show that the average latency to generate ECDSA signatures, over 256 bit prime fields, is 0.4 ms. Using

a dedicated hardware module, the results indicate that even modest computing resources can support generating 6742 signatures per second [181]. Note that RSUs will be provided with higher computational power; thus, generating thousands of chaff CAMs will not incur excessive workload on the RSUs.

In order to ensure the timely generation of chaff messages, RSUs retain a pool of ready-to-use pseudonyms. We provide an estimate for the size of this pool, by measuring the maximum propagation delay between adjacent mix-zones. Using the arriving real vehicles and hence the number of chaff vehicles, we can compute the required chaff pseudonyms. According to our results in Table 6.4, a pool size of five chaff pseudonyms ensures the timely availability of ready-to-use pseudonyms.

For safety applications to operate correctly, CFs are distributed to vehicles in transit between adjacent RSUs. We provide an upper bound for their size to assess the storage and communication cost. After a full-day simulation, we found a maximum of 175 and 165 simultaneously active chaff pseudonyms across all mix-zone for $R^{MIX} = [100, 150]$. After examining the mix-zone placement, we can explain this peak through a lower mix-zone density. Given the constraint of non-overlapping zones, mix-zones are placed sparser across the same area causing longer unencrypted segments between zones. The results are longer chaff traces and hence simultaneously active chaff pseudonyms. Even though a CF containing these pseudonyms is sufficient for safety applications, it requires to be continuously updated, leading to an increased overhead during epidemic propagation. As mitigation, a larger number of chaff pseudonyms can be proactively included in the CF. For example, using the maximum simultaneous active chaff pseudonyms, i.e. 176, we can compute that 316,800 chaff pseudonyms are sufficient to provide a CF that needs to be updated every 30 minutes. This amounts for a CF with 3.63 MB in size at a false positive rate of $p = 10^{-20}$ [177]. Given a data rate of several Mbit/s for modern IEEE 802.11p interfaces [119], this means CF updates do not pose a significant communication overhead. With respect to storage requirements, given that the size of a pseudonym is less than 3.5 KB [120], half an hour worth of credentials requires 1.1 GB of storage at an RSU.

6.7 Physical Layer Tracking

Even though encryption can hide the content of CAMs, it cannot conceal their transmissions on the wireless medium. Assuming that data-link level identifiers, i.e. MACs and IPs, as well as pseudonyms are changed covertly, we are interested in whether wireless transmission characteristics can be exploited to compromise pseudonym change. At an intersection, we conducted experiments to acquire raw In-phase and Quadrature (I/Q) samples from transmissions of a NEXCOM OBU using SDRs to explore how these could be used to link a vehicle's pseudonyms.

While radio frequency device fingerprinting [182] is a related concept, our approach differs in the sense that we capture location specific characteristics. Instead of building a database with a fingerprint of every victim device, we extract transmission features that can potentially affect any OBU. This way, the attack becomes a persistent threat, since the required ground truth can be acquired once per intersection instead of once per vehicle.

6.7.1 Methodology

We consider a passive eavesdropping attacker who uses SDRs with at least two antennas to capture I/Q samples of CAM transmissions. Based on these data streams, a model that maps radio signal properties to a particular region of the intersection is created. The attacker uses signal characteristics caused by multipathing, reflections, the OBU's antenna's radiation pattern, and distance between sender and receiver to track the victim's vehicle as it traverses a CMIX. Labelled data can for example be acquired using visual observations that are correlated with signal reception times or, to get a more precise ground truth, the attacker could profile the target intersection by traversing it with his own vehicle while recording GPS coordinates with timestamps.

6.7.2 Experimental Setup

We installed a NEXCOM vehicular OBU (Figure 6.9a) inside the experiment vehicle and mounted its antenna on the roof (Figure 6.9b). This vehicle fulfilled two roles;

it acted as the attacker's accomplice by providing its location as ground truth to create a model of the intersection, and as the victim sending encrypted CAMs. During our experiments, the vehicle traversed the intersection using all possible combinations of entry and exit. These are labelled as sectors A, B, and C in Figure 6.10. We implemented a library to send UDP packets via the IEEE 802.11p interface of the OBU which was used to broadcast CAMs at a frequency of 10 Hz when the experiment vehicle is moving. The payload we recorded for our analysis was the current time and GPS location of the vehicle. A second vehicle, the red cross in Figure 6.10, equipped with two Ettus N210[6] Universal Software Radio Peripherals (USRPs) (Figure 6.9c) served as the eavesdropper.



(a) **(b)** **(c)**

Figure 6.9: Vehicle with NEXCOM OBU (a) & (b) and two MIMO synchronised Ettus N210 USRPs used to acquire I/Q samples (c).

The USRPs were synchronised using a Multiple-Input and Multiple-Output (MIMO) cable and connected to a laptop which stored the I/Q samples. To identify data packets, we used a custom GNU radio block[7] which allowed to demodulate the samples and acquire their payload, i.e. timestamps and GPS coordinates.

[6]https://www.ettus.com/product/details/UN210-KIT
[7]https://github.com/bastibl/gr-ieee802-11

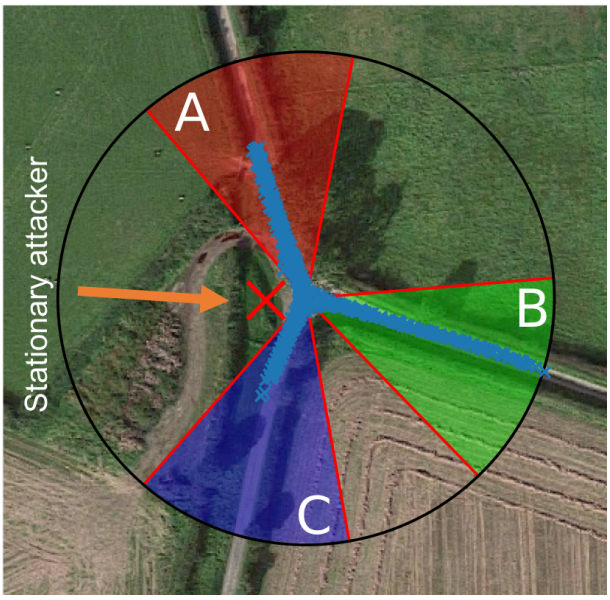


Figure 6.10: Our attack scenario; each road of the intersection is assigned to a sector (A, B, and C). A blue cross indicates the location of a received packet. The stationary attacker is shown with a red cross.

6.7.3 Data Processing

We separate the acquired data such that we can determine the entry-exit sectors, I/Q sequence, distance to the intersection centre, and GPS coordinates for each profiling CAM. From the I/Q samples, we compute the following features: mean transmission power, mean phase shift, and the correlation of the power received by the two eavesdropper antennas. The resulting feature vectors are used in two stages; first to determine the sector of origin, and second to estimate the victim's distance to the turning point of the intersection, i.e. the centre.

Sector classification

In this phase we use a supervised random forest classifier with three classes to determine the sector from which the CAM was emitted. For training, each feature vector is labelled with the respective sector label.

Distance estimation

To determine the sender's distance from the centre of the intersection, we use random forest regression. For training, we provide the distance between the intersection's

centre and the victim's vehicle as label.

6.7.4 Results

After a 5-fold cross validation, our sector classifier achieves an accuracy of 93% on average. For feature importance, a weight of 68% was assigned to the phase shift, 29% for the transmission power, and 3% to the power correlation. Intuitively, the phase shift can be used to capture a signal's angle of arrival, therefore it remains mostly unchanged as a vehicle approaches the intersection in a straight line and hence characterises the sector a vehicle is located in.

To assess the performance of the distance regression, we measure the mean absolute error. Using 5-fold cross validation, this error was found to be 10 meters on average. As for feature importance, 28% was assigned to phase shift, 70% to transmission power, and 2% to power correlation. These results reflect how the transmission power is more descriptive for distance estimation as it decreases with increasing displacement between sender and receiver. Overall, power correlation does not contribute significantly to classification decisions.

6.8 Discussion

Trusted and Cooperative OBUs

Similar to the work which initially proposed the CMIX approach [146], our adversary model is only concerned with external eavesdroppers. We argued that the openness of VANETs and their proximity to the WiFi spectrum makes such attacks plausible and worth investigating. Yet, before this pseudonym change strategy can be deployed, the threat posed by an internal attacker has to be investigated. There are two ways an internal attacker can tamper with the system. First, he can simply not participate in the CAM delegation and hence take out the traces of some of the chaff vehicles. If a single OBU misbehaves like this, the impact on privacy would be rather small since it is limited to compromise the link between two mix-zones. Second, colluding attackers could collect mix-zone keys to link pseudonyms by accessing the CAMs within the encrypted regions. However, to obtain these keys,

valid VANET credentials are required. This means, an attacker has to buy a set of vehicles or compromise them with malware. While the latter could be mitigated through HSMs, the former might be feasible. A potential mitigation strategy could be rotating mix-zone keys. If the mix-zones' encryption keys are frequently changed, the resources required by the attacker can be increased since he would have to position an OBU at each mix-zone to witness key updates and collect CAMs.

Physical Signal Tracking

Through our initial study in Section 6.7, we gave an outlook about the feasibility of using physical layer properties to track a vehicle despite the use of encrypted CAMs. Although the classification results seemed promising for an attacker, it is necessary to implement an actual attack algorithm before drawing conclusions about the severity of such attacks. Two techniques that come to mind to achieve this are Multiple Hypothesis Tracking [127] or Nearest Neighbour Probabilistic Data Association [21]. Additionally, experiments featuring several vehicles are required to account for interference and other vehicle related signal property changes that could not be recorded by our setup. Lastly, the placement of an eavesdropper device also has to be investigated. In our experiment we placed the attacker's vehicle almost at the centre of the intersection. In a real scenario, this might either not be possible due to physical constraints of the environment or it might at least not be easy to conceal so that an attacker can be spotted and prosecuted. Even, if we assume that an attacker's device could be placed covertly, recent research by Chaman *et al.* [183] suggests that it may be possible to detect an eavesdropping device based on the electromagnetic radiation of the signal processing equipment.

6.9 Summary

In this chapter, we proposed a scheme for private pseudonym change that offers a constant level of privacy independent of the operation area, mix-zone encryption radius, time of the day, and vehicle arrival rates. Based on the idea to provide a crowd to hide in, we introduce chaff vehicles when needed and refrain from

their dissemination otherwise. Our scheme ensures that OBUs are provided with sufficient information to distinguish between real and fictive vehicles in order to preserve the correct operation of safety applications. We analysed the security of our protocols and showed the privacy gained as well as the performance overhead imposed by our system. In comparison to a previously proposed CMIX scheme, without significant additional overhead, our scheme was able to improve user privacy up to 76%, independent of the underlying traffic conditions. We gave an outlook as to how physical layer tracking poses a threat to encryption based pseudonym change strategies and discussed the implications of our results.

As signing material for this chapter, we assumed vehicle specific ECC credentials that are frequently changed. Through pseudonymisation, an additional abstraction is added to publicly transmitted certificates which is supposed to protect the identity of drivers as well as owners. This abstraction can be removed by an RA to provide law enforcement, i.e. the police, with a lead to prosecute traffic offenders. However, except some cases when an individual is the owner and sole user of a car, the owner of a vehicle and the driver is not necessarily the same person. To overcome this limitation, we explore the idea of driver instead of owner specific signing material in the next chapter. We use biometric information to tie a driver's identity to VANET signatures that fulfil the requirements discussed in Chapter 4 while providing conditional anonymity.

Where large sums of money are concerned, it is advisable to trust nobody.

— Agatha Christie

7

Anonymous Driver-centric Message Authentication

Contents

7.1 Driver-centric Message Authentication 141
7.2 System Proposal 142
7.3 Concept and Assumptions 143
7.3.1 Application Scenario: Strong Driver Authentication . . 143
7.3.2 System Model 145
7.3.3 Adversary Model 146
7.4 Message Authentication Protocol 147
7.4.1 Underlying Principle 148
7.4.2 Enrolment 150
7.4.3 Bionym Acquisition 151
7.4.4 Message Authentication 152
7.5 Security Analysis 153
7.5.1 Passive Eavesdropper 153
7.5.2 Bionym Linkability 154
7.5.3 Active Manipulation 154
7.5.4 Relay Attack Mitigation 155
7.6 Performance Evaluation 156
7.6.1 Biometric Performance 156
7.6.2 Network Performance 159
7.7 Discussion 161
7.8 Summary 162

In the previous chapter we showed how pseudonyms can be used to improve

driver privacy by breaking a journey down into smaller, unlinkable segments. The signing material we considered tied the identity of a vehicle's owner to its messages and discouraged malicious behaviour under the penalty of prosecution. Yet, if driver and owner are not the same, it is difficult to identify the person causing an accident or committing a traffic offence. This is particularly relevant for increasingly popular car sharing schemes and Transportation as a Service (TaaS) where vehicles are owned by mobility providers.

In this chapter, we propose a message authentication scheme which incorporates biometric information into message signatures, making each message traceable to the actual driver. This enables accountability and exclusion from the network on a per-individual basis. Our scheme satisfies the requirements for VANET signatures stated in Chapter 4 while at the same time providing fine grained non-repudiation. We implemented a prototype to demonstrate the feasibility of acquiring a driver's body impedance as one possible instantiation of our scheme. This modality comes with the advantage that it can be measured while a driver is holding the steering wheel. Our evaluation features a user study with 33 subjects who were asked to steer a truck under simulated driving conditions. We run simulations on a realistic road network to assess the revocation delay of our scheme. This chapter is based on our work in [184].

Disclaimer: Due to the collaborative nature of the work this chapter is based on, small text segments overlap with the DPhil thesis submitted by Marc Roeschlin. Overall, our individual contributions can be separated into four categories: VANET, biometrics, cryptography, and evaluation related. The background knowledge about VANETs, the entailed security and privacy requirements as well as the tools to simulate vehicular networks were contributed by myself. Marc provided his expertise with biometric traits as user credentials and the necessary machine learning methods. We equally contributed to both, the development of the presented protocols as well as the security and user based evaluation. We built the prototype devices in our shared laboratory in the Computer Science Department at Oxford.

7.1 Driver-centric Message Authentication

While the idea of incorporating biometric information in message signatures is not entirely new, previous work does not fulfil the requirements for VANET message authentication schemes. We are going to discuss the approaches in [185–187] which pioneer the idea briefly.

The authors of [185] propose to enhance user authentication in VANETs using face and fingerprint biometrics. They layer the two biometric modalities to derive a key and encrypt messages using the Exclusive-Or (XOR) operation. The computational overhead and authentication time of their approach is evaluated in a scenario with roadside units, authentication servers, and up to 100 simulated vehicles. However, since the protocol uses an XOR cipher it allows an adversary to recover the biometric material via a chosen plaintext attack. Given that the structure of CAMs is publicly available, an attacker can easily infer the payload of a message using external information such as GPS coordinates of the vehicle's current location and its state, e.g., velocity and heading. This violates privacy by facilitating tracking but also enables the attacker to sign messages on behalf of the victim which conflicts with the non-repudiation requirement. Furthermore, the absence of revocation and traceability combined with a fixed message size renders the protocol unusable.

The scheme in [186] offers mutual authentication using vehicle movement. The authors suggest a Keberos-like method by replacing symmetric with biometric encryption. Without giving a description of how the biometric information is embedded into the protocol, this scheme is not ready for use. Moreover, given the efforts that have gone into the development of a VPKI architecture with respect to scalability, security, and privacy, the deployment of this approach seems unlikely.

The biometrics-based anonymous authentication presented in [187] uses biometric encryption and suggests temporary MAC addresses for privacy protection. Every pair of vehicles establishes a separate communication session which makes message broadcasting obsolete, but at the same time introduces an overhead which is quadratic to the number of communicating vehicles. Furthermore, establishment and exchange of symmetric keys are not specified. In a simulation, the authors

evaluate passenger privacy independent of the proposed biometric encryption scheme. Its applicability under real-world conditions remains unknown.

7.2 System Proposal

By tying the cryptographic material used to sign messages to a vehicle's owner, she can be automatically held accountable for the transmitted messages. This is particularly challenging for company cars, rental cars, and TaaS. However, exactly these applications are projected to show an annual growth of over 30%, as cities in Europe aim to make it unnecessary to own a private car in the next decade [188, 189]. Furthermore, it is the car sharing company's obligation to keep a record of each driver's identity they provide with a vehicle. Customers are thus required to disclose personal details, e.g. their passport, for every journey while they could otherwise remain anonymous if they owned a car themselves. Undoubtedly, a VANET infrastructure that could provide fine-grained message authentication while providing conditional privacy would allow for relaxation of privacy intruding procedures if it ensures that the driver can still be held accountable for her actions. Even though research and industry have identified the use of biometrics in the automotive context, such as for access control and immobilisers, existing proposals for biometric-enabled message authentication do not meet today's requirements for VANETs. Trying to remedy those deficiencies, we propose a new approach based on a commitment scheme that uses anonymised biometric measurements. We summarise the contributions made in this chapter as follows:

- Using repeatedly measured biometric traits during vehicle operation, we propose a message authentication scheme that implements the issuance and revocation of key material on a per-individual basis. This allows to bind the identity of the actual driver to VANET messages.

- We instantiate our solution using body impedance as the biometric modality and conduct a user study with 33 subjects using a prototype set-up.

- We provide a thorough analysis, to evaluate the security and privacy properties of our protocols and run simulations on a real-world road network to assess revocation delay.

7.3 Concept and Assumptions

For our message authentication scheme, we propose a modification of the signing material acquisition procedure. In this section, we discuss the current state of personalised car security and describe how our protocol complements existing approaches. Afterwards, we define the system and adversary model considered in the remainder of this chapter.

7.3.1 Application Scenario: Strong Driver Authentication

Many car manufacturers offer passive key-less entry and start systems. Similar to the user authentication presented in Chapter 3, these systems grant access to a driver's vehicle after an authentication protocol between vehicle and token was successful. However, a hardware token cannot guarantee identity verification as it might be given to or stolen by another person, similarly to how a car key can be shared.

Thus, in an attempt to personalise access control, car makers have started to adopt biometrics for immobilisers and payment systems. Using a biometric trait adds its non-transferability property that cannot be provided by a hardware token. If acquired seamlessly, biometric readings allow uninterrupted driver verification throughout a journey. Otherwise, if only verified once per trip, additional measures are needed to guarantee the continuity of the driver's identity. For example, her identity would have to be invalidated when an open door or window is detected.

Our message signature scheme uses commitments, called *bionyms*, which are unique but anonymised identifiers designed to authenticate VANET messages. Bionyms are derived from a driver's biometric characteristics following the steps shown in Figure 7.1.

In order to acquire bionyms, the driver has her biometric characteristic measured by a sensor built into the vehicle. Next, a request containing an anonymised

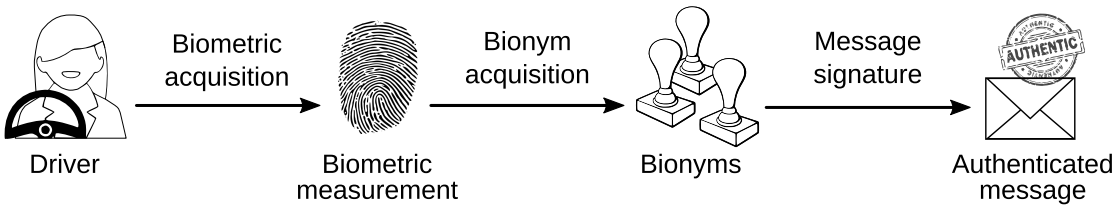


Figure 7.1: Data flow: A driver requests bionyms by providing her biometric measurement. The bionyms are anonymised signing material that is used to generate VANET message signatures.

biometric measurement is sent to a trusted authority to verify the driver's identity. If successful, bionyms are issued and stored in the vehicle's OBU. These carry the anonymised biometric, are unlinkable, and have non-overlapping expiration dates to prevent Sybil attacks. To ensure the availability of valid bionyms, an OBU periodically refills its credential repository using the VANET infrastructure. Since the rate of bionym requests depends on the number of bionyms provided during one acquisition and their expiration date, we can control the re-validation frequency of a driver's identity with these parameters.

Our proposed solution uses a combination of a possession-based and biometrics-based mechanism to prevent impersonation while providing privacy to the driver. Therefore, we require the possession of a token, e.g., a smart watch, wristband, or smart phone, and a biometric modality. The trust assumptions towards the token are minimal, it functions as a storage for a blinding value used to anonymise biometric information. If not combined with the driver's measurement, the blinding value is useless to an attacker.

We do not consider a solution where the authentication process is executed on an intermediary device: If the device is wireless, special provisions are required to mitigate relay attacks, and if the device is, e.g. a smart-card, the vehicle needs an interface in addition to the biometric sensors. Furthermore, the device would have to be fully trusted and tamper-proof such that it can store biometric information in a secure way.

7.3.2 System Model

Without the loss of generality, we abstract the VPKI described in Section 4.5.4 and replace the individual CAs with a central Trusted Authority (TA) for the remainder of this chapter. The separation into multiple authorities can be implemented analogue to a scheme like SECMACE [123]. The simplified network model is shown in Figure 7.2. It includes a TA, RSUs, and OBUs.

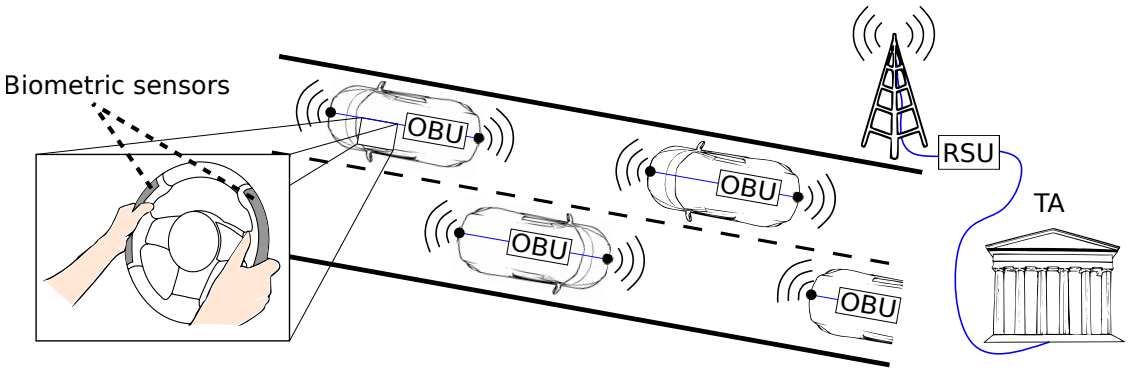


Figure 7.2: System model with the three VANET entities: TA, OBUs, and RSUs. RSUs relay information between OBUs and the TA. Vehicles can acquire a driver's characteristics via sensors embedded into the steering wheel.

Trusted Authority

A driver enrols at the TA. This process is supervised and can be combined with the issuance of a driving license. Once her identity is verified, the TA stores information necessary to provide an OBU with driver-specific *bionyms*. To protect this acquisition process, every vehicle can establish a secure communication using TLS and the TA's public key via RSUs. Additionally, the TA maintains a mapping between real identities and bionym acquisition information for non-repudiation.

Bionym

A bionym is a unique identifier which allows authenticating messages sent by a vehicle while being operated by a certain driver. To compute message signatures, the bionym has to be combined with the driver's biometric information. We define a bionym's life-cycle in three stages: *acquired*, *in use*, and *expired*. In case no bionyms

are available, a vehicle has to acquire them from the TA. Thereafter, these can be used until they expire, thus are no longer accepted by other vehicles.

Driver

A driver is required to enrol at the TA by providing her biometric characteristic and a second factor, e.g. a smart phone. Once the biometric measurement is stored, the ability to acquire bionyms is granted. This can be exercised using any OBU as long as the driver provides biometric measurements and carries the second factor with her.

Vehicle

A vehicle's OBU includes a HSM that enforces the bionym life-cycle, securely stores private keys, and erases excess bionyms at the end of a vehicle's journey. OBUs are connected to the internal bus of vehicles and can access peripherals like biometric sensors. Before broadcasting a message, an OBU generates a signature using a bionym and the driver's biometric measurement.

7.3.3 Adversary Model

We extend the model used in Chapter 6 by also considering an attacker who aims to manipulate the traffic flow by inserting messages with forged signatures. Moreover, we introduce an impersonator who tries to circumvent biometric recognition, either physically or by reusing biometric information from captured messages. Independent of the attacker's goal, we assume computations are limited to polynomial time and the computational Diffie-Hellman assumption holds within a properly constructed cyclic group.

Tracking

A passive adversary can observe VANET messages and try to exploit the location information in CAMs to reconstruct a victim's journey. If the location of the victim can be derived for multiple points in time, i.e. an entire journey, chances to identify an individual are very high [141]. Generally, a tracking adversary is considered successful if he can derive a static identifier directly from the messages or if the

observed trip segment identifies the victim. We assume that he neither has access, e.g. by using malware, nor is able to follow his target, e.g. by maintaining a network of cameras, which would make tracking trivial.

Traffic Manipulation

ITSs use traffic information to optimise throughput and reduce congestion within the road network. Fabricated messages can be used to adversely affect such systems, for example by reporting non-existing accidents or ghost vehicles. This creates fictive congestion which could force drivers to slow down or even take a detour. An attacker can use this to facilitate car-jacking, robberies, or cause economic damage, e.g. in case of toll roads and competing infrastructure providers.

Impersonation

The adversary can attempt to impersonate a driver to send VANET messages on her behalf. If he is physically present in a vehicle, he can present his own or forged biometric material to the OBU. Otherwise, he can try to forge signatures by reusing captured bionyms. Either way, we assume that the adversary has not obtained a valid biometric reading of the victim and can not break the HSM of the OBU.

7.4 Message Authentication Protocol

The technique we use leverages properties of the integer commitment scheme proposed by Damgard *et al.* [190]. Signatures are created as non-interactive proofs of knowledge that certify the validity of the biometric measurements provided by the driver. These allow recipients to verify the authenticity and integrity of VANET messages. To enable the use of a pseudonym change strategy as presented in the previous chapter, our protocol allows OBUs to acquire a set of multiple bionyms. These are indistinguishable from random and cannot be linked based on their digital representation. Hence, an attacker has to re-identify the victim's vehicle every time the OBU switches credentials. Randomising the biometric information also mitigates bionym leakage since it prevents the extraction of the original biometric measurement.

This is important as biometric features cannot be changed once revealed to an attacker. We define the notations for the remainder of this section in Table 7.1.

Table 7.1: Summary of protocol parameters and notations.

$H(\cdot)$	Cryptographic hash function
$h \leftarrow \mathbb{Z}$	Random number h drawn from $\mathbb{Z}$
$\mathsf{KeyGen}(\cdot)$	Biometric key derivation function
$[\cdot]_{K_X^+}$	Encryption and HMAC based message integrity
$\langle\cdot\rangle_{K_X^-}$	Signature using ECDSA with ECC credentials of X
$\mathsf{PRNG}(\sigma)$	Cryptographically secure pseudorandom number generator seeded with δ

7.4.1 Underlying Principle

In order to use Damgard's scheme, a onetime set-up procedure is required to determine the public parameters of the scheme. Therefore, the TA constructs an algebraic multiplicative group G such that computing roots of random elements in G is computationally infeasible. The public parameters g and h are determined as follows: $h \in G$ is a randomly picked element and $g = h^\omega$ for a random secret ω.[1] The description of G, g, and h are announced to all vehicles in the TA's province and $g \in \langle h \rangle$ is proven without revealing ω via a Schnorr signature based statistical proof of knowledge [191].

In addition to g and h, the TA's public key K_{TA}^+ and the two algorithms $\mathsf{BioSign}$ and $\mathsf{BioVerify}$ are assumed to be public. These algorithms implement the signature generation and verification of our commitment scheme as shown in Algorithms 1 and 2.

In order for $\mathsf{BioSign}$ to authenticate message payload p, three auxiliary inputs are needed: commitment c, biometric key k, and blinding value γ. The signatures generated by Algorithm 1 consist of the tuple $\sigma = (d, v, u)$. Which kind of commitment is used by $\mathsf{BioSign}$, depends on the recipient of the message: When interacting with the TA, it is necessary to identify driver D for non-repudiation,

[1] For a detailed description of the parameter initialisation, we refer the reader to [190].

Algorithm 1: BioSign - Generate signature σ for payload p under commitment c with biometric key k and blinding factor γ. Signature σ contains key k only in blinded form.

Global: bases g and h, cryptographic hash function $H(\cdot)$
Input : commitment c, biometric key k, blinding factor γ, payload p
Output : signature σ
$y \leftarrow \mathbb{Z}\,, \quad z \leftarrow \mathbb{Z}$ //draw randomly
$d = g^y \cdot h^z$
$e = H(c, d, p)$
$u = y + e \cdot k, \quad v = z + e \cdot \gamma$
return $\sigma = (d, v, u)$ //pack signature

hence driver specific commitment C_D is used. When signing an anonymous message, providing a commitment in the form of a bionym b_i is sufficient. To make b_i non-static it contains a random value δ_i which prevents association with the driver's identity. The commitments and their corresponding blinding factors γ are set as shown in Equation 7.1 and 7.2.

Algorithm 2: BioVerify - Verify signature σ of a payload p using commitment c.

Global: bases g and h, cryptographic hash function $H(\cdot)$
Input : commitment c, signature σ, payload p
Output : *True* or *False*
$(d, v, u) = \sigma$ //unpack signature
$e = H(c, d, p)$
if $g^u \cdot h^v = d \cdot c^e$ **then**
| **return** *True* //Equation 7.3 holds
end
return *False*

In either case, a message's signature σ can be verified as shown in Equation 7.3 using BioVerify, if and only if σ was generated with BioSign over valid inputs.

$$\gamma = s, \qquad c = C_D = g^k \cdot h^s \ : \text{interacting with TA} \tag{7.1}$$

$$\gamma = \delta_i + s, \quad c = b_i = C_D \cdot h^{\delta_i} \ : \text{anonymous messages} \tag{7.2}$$

$$g^u \cdot h^v = \ g^{y+ek} \cdot h^{z+e\gamma} = g^y \cdot h^z \cdot (g^k \cdot h^\gamma)^e = d \cdot c^e \tag{7.3}$$

In the next section, we give a detailed description of the parameters used in the algorithms and explain the three phases of our protocol: *Enrolment*, performed once

for every driver, *Bionym Acquisition*, executed periodically when new bionyms are required, and *Message Authentication*, invoked whenever a message is transmitted.

7.4.2 Enrolment

Before a vehicle can sign messages under a driver's identity, the driver has to be registered with the TA by providing her identity D, a biometric measurement q, and a secret s stored on a token (see Figure 7.3). An officer at the TA ensures that D is associated with a genuine biometric measurement q by following the intended acquisition procedure. Afterwards, the biometric key $k = \mathsf{KeyGen}(q)$ is derived. $\mathsf{KeyGen}(\cdot)$ is a key derivation function that transforms the biometric characteristic into a cryptographic secret in an irreversible way similar to a hash function. The resulting secret appears indistinguishable from random and does not allow the inference of any biometric information [192].

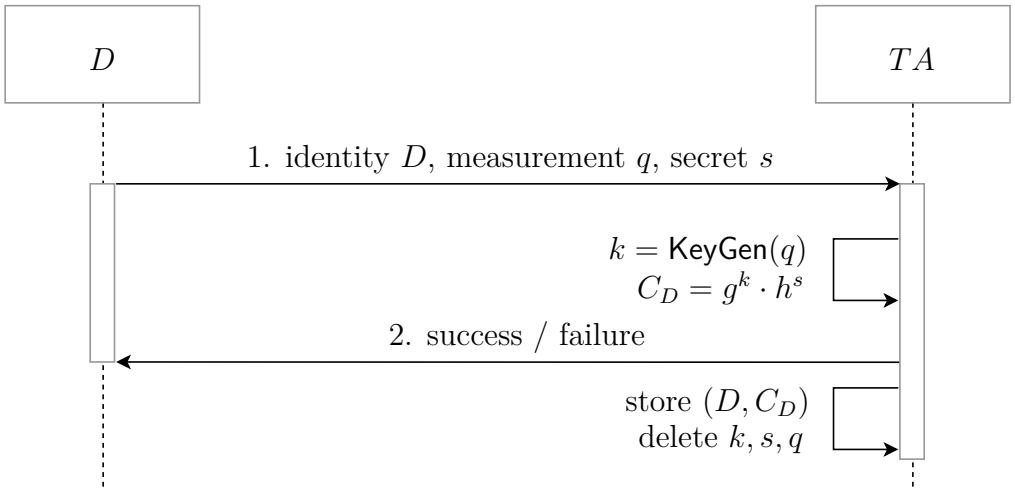


Figure 7.3: Enrolment of driver D at the TA. After her identity is confirmed, the enrolment commitment C_D is established and stored to verify D during bionym acquisitions.

After key k is generated, the TA creates an enrolment commitment $C_D = g^k \cdot h^s$ that encapsulates k and thereafter erases s, k, and q. Finally, the driver is notified about the outcome and the tuple (D, C_D) is stored. When interacting with the TA, the OBU computes C_D on-demand to authenticate as D.

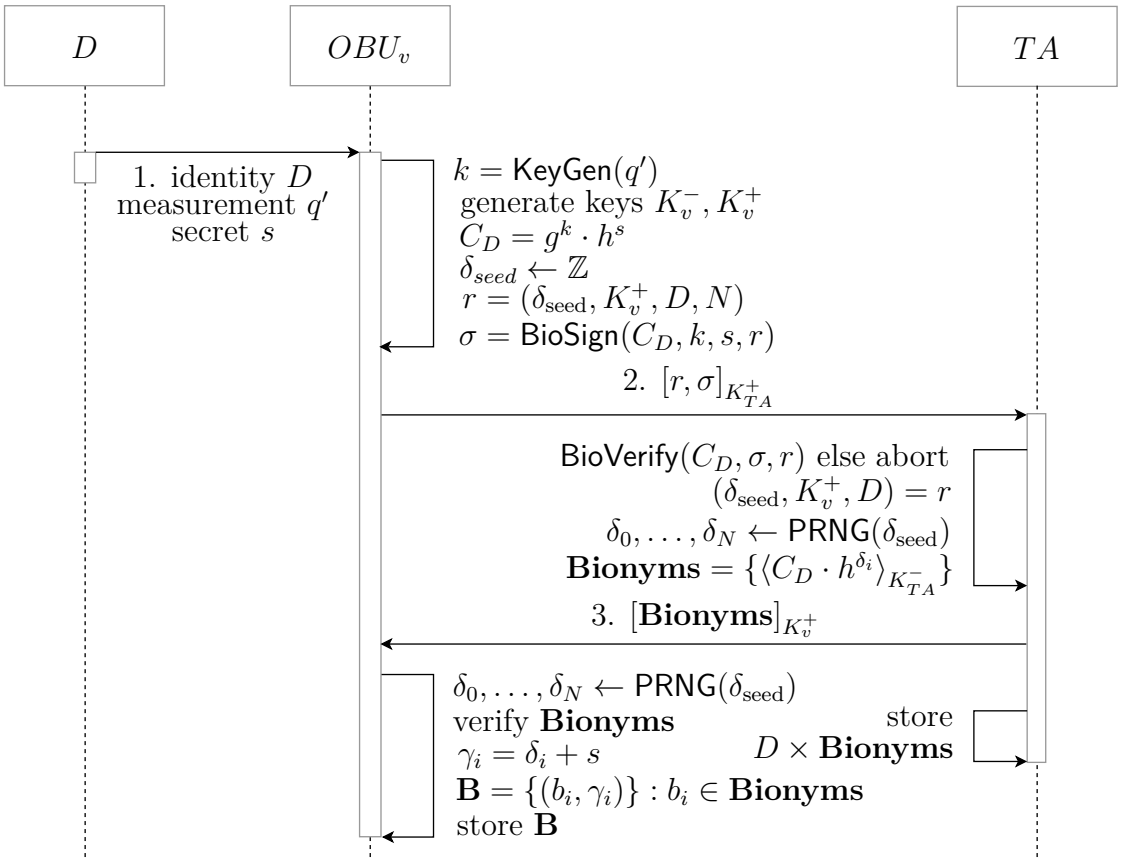


Figure 7.4: Acquisition of N bionyms: A driver provides her biometric measurement before OBU v requests and stores a batch of bionyms from the TA.

7.4.3 Bionym Acquisition

Bionyms are acquired from the TA in batches. Figure 7.4 shows this process which starts with the driver providing her secret s and a new biometric measurement q' to the vehicle. After the OBU v has extracted the biometric key $k = \mathsf{KeyGen}(q')$, it uses a fresh asymmetric key pair K_v^-/K_v^+ for bionym retrieval. While the biometric has to be provided for each acquisition, s can be stored on the OBU for the duration of the trip. In case the driver prevents the acquisition of q', the vehicle will not start if stationary or not be able to acquire new bionyms and hence be traceable if already driving. The tuple of public key K_v^+, a random seed δ_{seed}, the number of bionyms N, and the driver's identity D form the bionym request r. For authenticity, this request is signed under commitment C_D, key k, and secret s using $\mathsf{BioSign}$. We note that for the generation of N bionyms, N randomisation values are needed. To keep the

communication overhead small, both entities, v and TA expand δ_{seed} to a sequence of N numbers using a cryptographically secure pseudo-random number generator. Upon receiving an acquisition request, the TA loads the commitment C_D for the claimed identity D and authenticates the request using BioVerify. If successful, δ_{seed} is expanded to the randomisation values δ_i to compute the bionyms $b_i = C_D \cdot h^{\delta_i}$ and sign them under the public key K_{TA}^{+}. The TA stores the real identity D with the issued bionyms for non-repudiation. Afterwards, the TA returns the set of bionyms encrypted with K_v^{+} to prevent an attacker from linking the enclosed bionyms. The vehicle verifies the bionyms using its own sequence of randomisation values δ_i. and stores the set of bionyms, $\mathbf{B}$, and blinding factors γ_i for later use.

7.4.4 Message Authentication

The authentication of VANET messages is depicted in Figure 7.5. Analogue to an acquisition request, the biometric key k, a bionym b_i, and blinding factor γ_i are required to sign a message m.

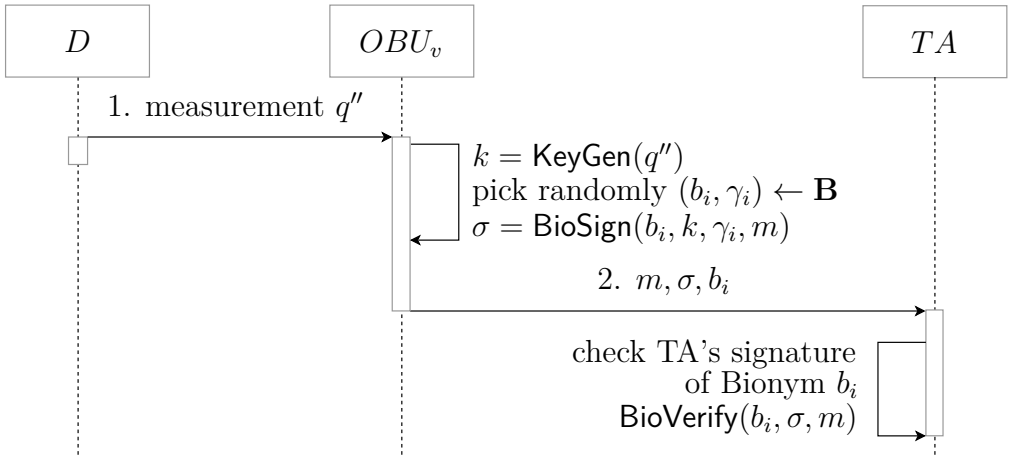


Figure 7.5: Authentication of message m: OBU v signs m and transmits the message, signature, and bionym over the network. Nearby vehicles $v^{\star}$ validate the bionym and verify the signature.

First, a new biometric measurement q'' is acquired from the driver to extract the biometric key k. To prevent linking attacks, the OBU chooses a fresh bionym b_i and the corresponding blinding factor γ_i. Signature σ is computed using BioSign and subsequently verified through BioVerify locally. If σ is not valid due to an incorrect

k, another biometric measurement is acquired to regenerate k. Finally, the message with signature and bionym is broadcast to nearby vehicles.

Similar to the TA during the bionym acquisition phase, receiving OBUs use BioVerify to check signatures. However, since the sender committed to the bionym at the TA, the bionym's authenticity can be confirmed using TA's public key K_{TA}^{+}.

7.5 Security Analysis

In this section, we analyse our system's resilience against protocol level attacks. We assume that the TA's public key K_{TA}^{+} is available to all OBUs and K_{TA}^{-} has not been compromised.

7.5.1 Passive Eavesdropper

An eavesdropper uses observed transmissions and tries to compromise secret s, biometric key k, or link consecutive messages of a vehicle for driver tracking and identification. In general, a driver's secret s is only used in its blinded form $\gamma_i = \delta_i + s$. Hence, the passive adversary cannot derive any information about its value. The seed δ_{seed} used in a cryptographic pseudo-random number generator to generate the randomisation values δ_i is encrypted with K_{TA}^{+}, thus protected when transmitted. Further, when overhearing a message, the adversary only learns its signature $\sigma = (d, v, u)$ and bionym b_i. The message m and value d are independent of k and γ_i, and therefore, neither can reveal information about the key or the secret. The values of $u = y + e \cdot k$, $v = z + e \cdot \gamma_i$ and the bionym b_i include k and γ_i in blinded form. Hence, the security of k and γ_i relies on the integer commitment scheme used for signature generation and verification. Damgard *et al.* [190] show that this scheme is unconditionally hiding under the root assumption. Therefore, any algorithm that could with non-negligible probability extract the biometric key k or γ_i from m, d, v, u, and b_i would break the computational Diffie-Hellman assumption.

7.5.2 Bionym Linkability

Unlike the TA, a passive adversary cannot resolve a bionym directly to an identity. However, he can leverage a bionym's uniqueness in combination with a vehicle's status messages to keep track of its location. As a defence mechanism, an OBU makes use of the fact that they are indistinguishable from random due to the randomisation values δ_i and changes bionyms frequently [120].

During bionym acquisition, a driver's identity D is transmitted encrypted with the TA's public key K_{TA}^{+}. In combination with the random value δ_{seed}, the resulting cipher text changes for every acquisition to prevent re-identification. To ensure that the return set of bioynms cannot be linked by the attacker, it is encrypted with the OBU's public key K_{v}^{+}.

7.5.3 Active Manipulation

A Dolev-Yao [33] adversary has full access to the transmission medium, but is not in possession of a victim's secret s or biometric key k. The goal of the adversary is to disseminate messages that are accepted by other vehicles and appear as if they originated from the victim or cannot be attributed to any real identity. We first focus on the case where the adversary replays or forges messages. Then, we analyse an attacker who aims to obtain or forge bionyms.

Message Forging and Replay

When modifying existing or creating new messages, the adversary must make the verifier accept the check $g^u \cdot h^v = d \cdot (b_i)^e$. This means that the signature $\sigma = (d, v, u)$ of message m has to satisfy $d = g^y \cdot h^z$ and $e = H(b_i, d, m)$ which is embedded in v and u. If the adversary cannot create a message that passes the verifier's check, message authenticity and integrity is preserved. According to the hiding property of Damgard's scheme, and assuming that $H(\cdot)$ is resistant against second-preimage attacks, an adversary would have to simultaneously guess either k and γ_i or y and z. While k and γ_i can be directly used for signature generation, compromising y and z allows to unblind u and v and obtain values to compute valid signatures. However,

the values y, z, and γ_i are picked uniformly at random, which means they can only be guessed with negligible probability. The difficulty of guessing k depends on the amount of uncertainty the biometric trait exhibits across a population and how well $\mathsf{KeyGen}(\cdot)$ can extract this entropy. We discuss estimates for this in Section 7.7.

Bionym Replenishment

If the adversary attempts to acquire bionyms on a victim's behalf, he has to send his own δ_{seed}, K^+_{Adv}, and the victim's identity D to create a new acquisition request. Signatures for bionym requests are constructed using secret s, commitment C_D and biometric key k; the adversary cannot generate a valid request without those parameters. Moreover, if C_D is unknown, i.e. it has not leaked form the TA, C_D can only be guessed interactively by using the TA as an oracle which can be hardened through rate limiting.

Bionym Forgery

When an adversary $\mathcal{A}$ is denied the acquisition of bionyms because his identity is revoked or not enrolled, he can try to forge bionyms. If the attacker can obtain a victim's biometric measurement q, he can use $\mathsf{KeyGen}(\cdot)$ to extract the biometric key k and choose a random secret s to compute $C_{\mathcal{A}}$. From this, he can construct a bionym $b = C_{\mathcal{A}} \cdot h^{\delta}$ using an arbitrary δ. Message signatures based on b can be validated with $\mathsf{BioVerify}(\cdot)$, but the attacker cannot sign b with the TA's key K^+_{TA} and thus, messages signed with this bionym will be rejected.

7.5.4 Relay Attack Mitigation

Like the CHAP protocol presented in Chapter 3, car keys and other security tokens can be used to unlock vehicles over a wireless channel. Similar to ZIA schemes, these systems implicitly assume the proximity of the token bearer and the vehicle based on their connectivity. In a man-in-the-middle attack [12], the adversary artificially relays messages between two devices to trick them into unintentionally completing the authentication process. When using bionyms, a driver is required to

be physically present and provide measurements through on-board biometric sensors. Hence, relaying messages is no longer sufficient and such attacks are prevented.

7.6 Performance Evaluation

We evaluate an implementation of our message authentication scheme by instantiating it with body impedance as the biometric modality. Since driving a vehicle involves holding the steering wheel, the unobtrusive acquisition of impedance measurements can be easily conducted through integrated conductive pads. Assuming that drivers only infrequently release the steering wheel, measurements can be captured whenever both hands touch these pads. This acquisition requires a current smaller than that of a standard battery and is nowhere close to having an effect on the driver's body.

Basic feasibility of body impedance for user authentication and biometric key generation has been shown in the literature [193, 194]. However, we apply this modality to a scenario where drivers move their hands and occasionally break contact to the biometric sensors. We therefore present a user study that assesses the applicability of body impedance in a simulated driving scenario.

7.6.1 Biometric Performance

We built a proof-of-concept acquisition system to capture body impedance. A schematic of our set-up is depicted in Figure 7.6. Two pads are attached to the sides of an off-the-shelf steering wheel usually used to control computer games. As soon as the wheel is touched, i.e. short-circuited by a human, one electrode emits a frequency sweep every 3 seconds. These sweeps range from 100 Hz to 100 kHz and are 300 ms long. After the signal has travelled through the human body, it is measured by the second electrode. The emitted signal and the measured signal are then correlated and transformed to the frequency domain in order to compute the complex impedance of the driver.

We conducted a user study with 33 participants (26 male, 7 female). The participants were aged between 25 and 40, and 94% of them owned a driving

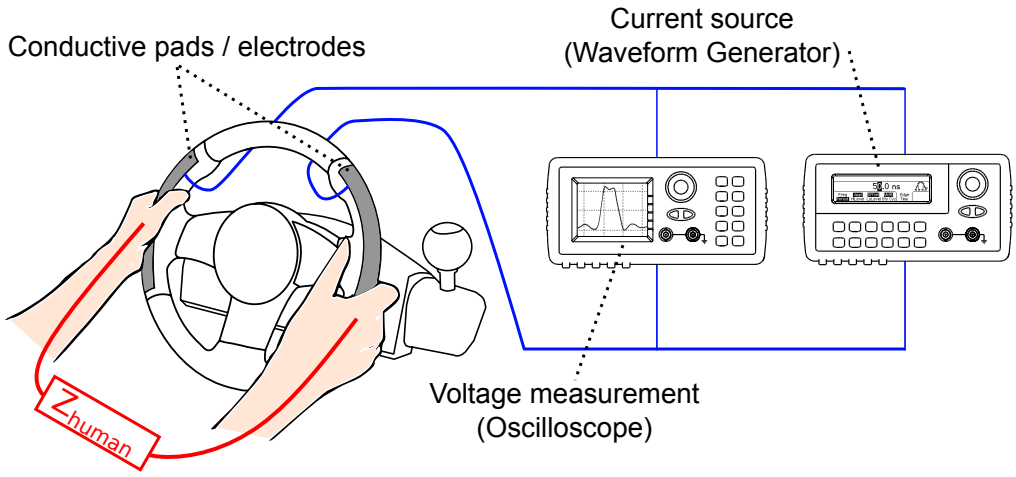


Figure 7.6: Measurement set-up: A waveform generator produces the frequency sweep which is received and recorded by an oscilloscope. When the human subject completes the electric circuit, the body impedance Z_{human} can be acquired.

license. The ratio between left-hand-side and right-hand-side drivers was 17:14. The participants were asked to sit and hold the steering wheel as if they were in a real car. To eliminate potential bias, the goal of the study was revealed after the fact.

We measured the participants' impedance in two configurations: while keeping the wheel still and while controlling a vehicle in a computer game shown on the screen in front of them. For each participant, we acquired 60 measurements over two 4 minute sittings. Having the participants steer a vehicle in first-person view elicits hand movements close to actual driving and the impact of hand positioning and steering motion can be analysed. On average, participants took 15 left and 12 right turns during their session; 95% of the acquisitions during turning succeeded while only 80% of the acquisitions during straight segments were successful. Participants were more reluctant to let go of the steering wheel when taking a turn than when driving straight which explains the difference in acquisition success. The voltage level used to acquire the measurements was set to 3 Volts, resulting in a current of 0.2 milli-Amperes, which is far less than what commercially available body fat scales use [195]. Our study was approved by the ethics review board, reference R55051.

Features and Classification

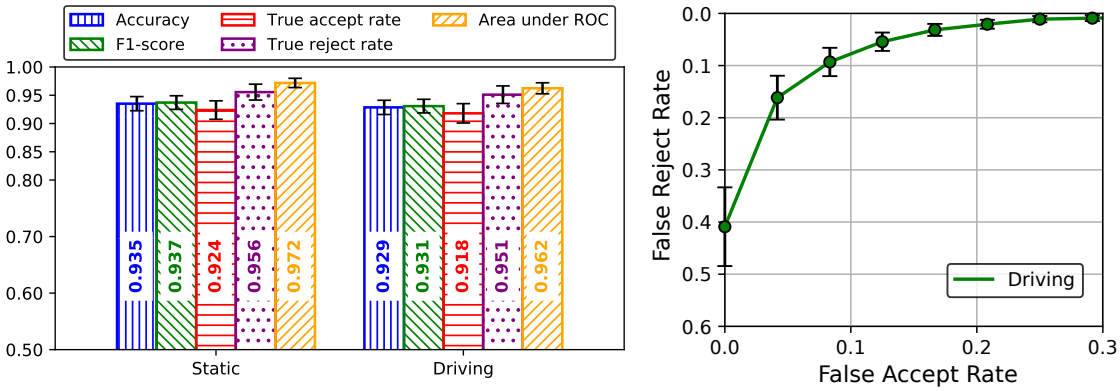
We calculate the normalised mutual information to identify the significant frequencies of the collected impedance measurements. Both, magnitude and phase, are most specific at lower frequencies, letting us choose the complex impedance at frequencies between 2 kHz and 76 kHz as the features for classification. A feature vector extracted from an impedance measurement is the array of its magnitude and phase values. We feed these arrays into a random-forest classifier that decides whether a measurement matches a reference reading of the claimed identity. A binary classifier was trained per participant by dividing the samples into two classes. One class comprises all the samples of the participant whose classifier is currently trained and the second class contains samples from the other participants.

Biometric Recognition and Impersonation

We evaluate the performance of biometric recognition based on body impedance using our prototype set-up. The obtained results reveal how likely an enrolled driver is recognised, i.e. a bionym request and a message authentication are successful. Further, the chances of an attacker impersonating a victim whose secret s has been compromised, that is when the security hinges on the biometric recognition only, can be estimated.

Figure 7.7a depicts recognition performance for the configurations *static* and *driving* using 5 different metrics. In order to minimise over-fitting, we conduct 5-fold cross-validation when obtaining these values. We further ensured that a ratio of 1:1 between measurements of an authorised driver and impersonation attempts was used for both, training and testing, which reduces effects of class imbalance. The values indicate that verification works well; performance for *driving* is on par with *static*. Even though most of the participants stated that they occasionally let go of the steering wheel when they drive their own cars, in our simulation, steering movement and the way drivers hold the wheel affect authentication only marginally. Unless the driver has a completely one-handed driving style, measurements can be captured in rapid succession and compensate for short interruptions. Moreover, as can be seen

in the receiver operating curve plot in Figure 7.7b, confidence for the false reject rate is high, implying a low chance for a systematic error. Hence, in a false rejection event due to noisy data or procedure errors, the reading can be re-acquired and is likely to be recognised. Therefore, a higher false reject rate can be tolerated to achieve a lower false accept rate and reduce probability for impersonation.



(a) (b)

Figure 7.7: Body impedance recognition performance for 5 different metrics (a) and receiver operating curve (b). Results are averaged over the test population and obtained through 5-fold cross-validation. Bars represent 95% confidence intervals.

7.6.2 Network Performance

Our scheme implements a refill-based revocation strategy. Every time an OBU attempts to acquire a new batch of N bionyms, the driver can be checked against a revocation list. Once an OBU's storage is depleted and refill is denied, the vehicle will not be able to produce valid message signatures. The duration until revocation comes into effect depends on the number of bionyms released per acquisition and their respective lifetimes. The impact of N on communication overhead is evaluated in the following.

Simulation Set-up

Similar to the evaluation of our CMIX scheme, we use SUMO [196] together with the LuST [173] scenario which provides 24h of vehicle traffic. The road network topology exhibits a mesh of small streets interconnected by arterial roads joining a highway

tangent to the city. Realistic traffic demand was modelled with two rush hours, one in the morning and one in the evening. The resulting simulation provides a large diversity in trip length, driving speed, and road types that resembles circumstances close to real-world traffic. As for the bionym change strategy, we assume a fixed bionym change interval as defined by the current standard [120][2]. The simulated bionym lifetimes are 50, 100, and 150 seconds, in line with literature in [197].

Revocation Delay

Passive, refill-based revocation does not require the TA to push additional information, e.g. revocation lists, to each node in the VANET. Hence, the delay between misbehaviour detection and exclusion from the network can be controlled using the acquisition batch size N and bionym lifetimes. In contrast to propagation-based revocation, which depends on RSU coverage and vehicle route dynamics, we can evaluate this delay directly. The results depicted in Figure 7.8 show the revocation delay relative to a vehicle's trip length.

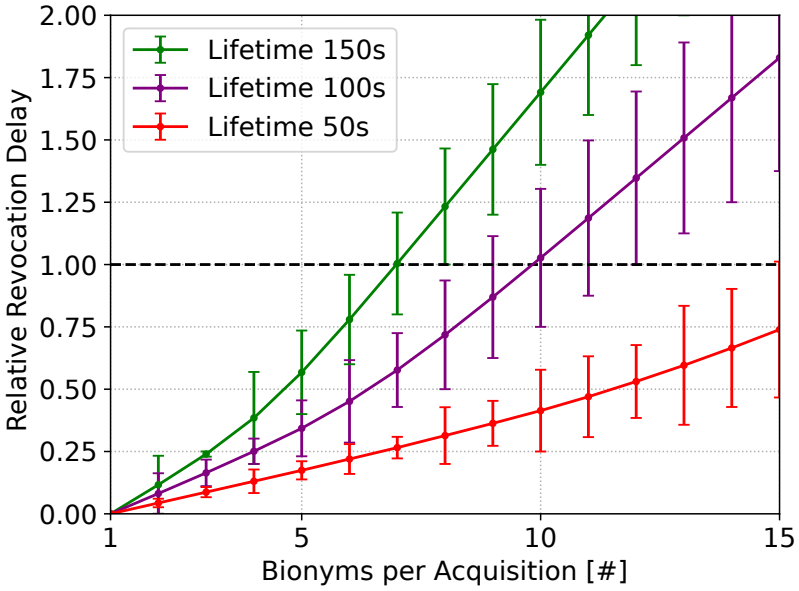


Figure 7.8: Average revocation delay for three different bionym lifetimes relative to trip length. A value of 1.0 means revocation after current trip. Error bars represent 95% percentiles across the vehicle population.

[2]We note that bionyms could also be integrated into our chaff-based CMIX scheme, however we opted to evaluate the message authentication scheme against the current standard to remove any bias towards our scheme.

A value of 1.0 indicates that the vehicle can just finish its current trip before being excluded from the VANET. It is evident that at a higher change frequency, i.e. a shorter credential lifetime, the same revocation delay is achieved with a larger N. For example, at a bionym lifetime of 50 s, an acquisition batch size of $N = 3$ leaves a vehicle with at most 10% of its journey before being revoked. Such a setting would require an acquisition every 150 s which is feasible according to our experimental setup that can provide biometric readings in under 3 s.

7.7 Discussion

Steering Behaviour

Our experiments showed that an acquisition takes less than 3 s. This can compensate for short interruptions if the driver touches the wheel at least from time to time. Yet, if the driver steers the car single-handedly[3], no measurement can be captured. This is a limitation of a body impedance based scheme which may not be the case for other recognition methods such as face recognition, eye movement tracking [199], driver posture [200, 201], or Electrocardiography (ECG) [202]. However, these methods come with their own drawbacks; a face recognition camera for example could record everything inside the car while other methods like ECG require inconvenient electrodes. Lastly, if self-driving vehicles become predominant, the question of liability has to be redefined; who is going to be held accountable the actions of such vehicles?

Biometric Key Generation

Our protocol uses biometric keys instead of a binary decision between accepting and rejecting an identity claim. While our experimental results do not allow to draw conclusions about the difficulty of guessing these biometric keys, they can adequately quantify the probability of impersonation. To obtain an estimate on the strength of keys derived from body impedance, we refer the reader to [194]. In

[3]Some countries advise or even mandate a driver to keep both hands on the wheel even for semi-autonomous vehicles [198].

said study, more than 50% of the generated keys exhibit at least 55 bits of entropy and 46 bits on average across a population. The reported false reject rate due to noisy readings is 8.6% and the success rate for impersonation is 1.9%.

Curious Trusted Entities

We replaced the VANET infrastructure with a single TA for simplicity. Obviously, having a single point of failure as well as a single point of trust is not ideal. But even if responsibilities are separated into different institutions, analogue to the architecture presented in Section 4.5.4 where PCAs could be replaced with bionym certificate authorities, special care has to be taken for entities that have access to the biometric material of the users. While passwords can be changed, a biometric trait cannot, hence the OBU and its HSM in particular have to be fully trusted by the user. Similarly, a user must be convinced that the TA indeed destroys the enrolment parameters after the commitment was generated. Establishing these trust relationships is an open challenge for human computer interaction researchers [203].

7.8 Summary

We presented a new scheme for message authentication in VANETs. Our approach enables the incorporation of biometric measurements into message signatures, transferring the responsibility from vehicle owners to the actual drivers. This provides features not found in existing schemes, such as the exclusion of individuals from the network and the protection of vehicle owners. If eventually adopted, our system makes the acquisition of personal data through rental companies and TaaS providers obsolete. At the same time, it prevents so called fronting, a type of insurance fraud where a parent registers for a vehicle on behalf of their child to avoid higher rates. Our protocols satisfy the requirements for VANET communication, i.e. authenticity, integrity, non-repudiation, conditional anonymity, and revocation while preserving the privacy of drivers and protecting their biometric measurements. We provide evidence for the feasibility of body impedance as a biometric modality in the form of a user study and measured the effect of credential provisioning on

revocation delay. Biometric recognition works sufficiently well and readings are available for signature generation in a timely manner. Although we argue that body impedance is particularly suited, the protocol is not limited to this biometric modality. Furthermore, our system can be integrated with any pseudonym change strategy, e.g. the CMIX scheme presented in the previous chapter.

I do the very best I know how,
the very best I can, and I mean
to keep on doing so until the end.

— Abraham Lincoln

8 Summary and Future Work

Contents

8.1 Summary of Results 165
8.2 Future Work 167
8.3 Final Conclusions 169

8.1 Summary of Results

This dissertation aimed to provide a systematic analysis of the security and privacy implications of location-aware mobile systems. Location awareness can provide convenience, for example, trigger a reminder when at the supermarket or turn on the light when a person comes home. Yet, as we showed, location as a contextual parameter has also been picked up by the developers of safety and security critical applications who have already started incorporating it.

We started by analysing how location information is commonly acquired using some external reference, showed the different threats these acquisition methods face and reviewed existing mitigation attempts. Next, we examined Zero-Interaction Authentication as a new paradigm for user authentication and presented how the implicit proximity assumptions of current implementations is susceptible to relay

attacks. To mitigate such attacks, we showed that a location can be described by the approach trajectory of a mobile device recorded with on-board inertial sensors. Using this as an additional factor, the dependency on an external reference could be eliminated.

While prover and verifier in a ZIA system do not rely on information from other parties, systems such as connected vehicles require situational awareness fuelled by a comprehensive pool of third party data for their safety applications. Therefore, in VANETs, vehicles broadcast their location, vehicle status, and other traffic relevant information. Recognising the security and privacy requirements of current VANET standards, we applied the idea of trajectories as proximity proofs to road networks and showed how vehicles driving in formation can authenticate the location claims of received messages. We provided a protocol allowing ongoing co-presence verification while preserving drivers' privacy.

Despite the merit of VANET messages for safety applications, these also have the potential to degenerate passenger privacy as they include the precise location of vehicles. Frequently broadcast CAMs not only happen to allow the tracking of individuals in real-time but also their identification based on commuting patterns. Using the idea of chaff traffic, we designed a CMIX scheme that enables vehicles to covertly change their identities, thus significantly enhancing passenger privacy.

As signing material necessary for such pseudonym schemes, researchers in the past mainly focused on purely cryptography based systems. These left the final mile in the chain of custody for traffic offences, i.e. to determine the actual perpetrator, to law enforcement personnel. Considering the intricate interplay of unique identifiers, message signatures, and context information that might pose a risk to a driver's privacy, we designed a message authentication scheme to fill this gap by integrating the driver's identity into message signatures. This not only offers an adequate way to prosecute people, but also bears the potential to reduce the amount of privacy sensitive data customers have to provide to rental companies.

While the solutions we provided might just be a first step towards the secure and private use of location information, we hope that they serve as a stepping stone

to spark future research efforts. In particular, because VANET standards are on the brink of completion and car makers are going to begin integrating them into the next generation of vehicles, soon. A comprehensive picture of the security and privacy implications of location information in mobile systems is required to inform policy makers who set up the rules for a *safe* and *private* future of transportation.

8.2 Future Work

Each of the systems presented in this thesis can be extended in several ways. In this section, we identify five immediate steps that either improve the systems as presented or foster a wider understanding of the security and privacy implications of location-aware mobile systems.

- **User specific trajectories:** So far we have used movement primitives to describe a device's trajectory through two-dimensional space. In Chapter 3 this represented the path taken by a user, in Chapter 5, we used this technique to describe the routes of vehicles. While we showed that trajectories are sufficient to assure the proximity and co-presence, respectively, additional features could be used to improve robustness and decrease false reject rates. Furthermore, in both scenarios, a longitudinal study during which users carry a device capable of recording inertial motion, e.g. a smart phone attached to a car mount, could be used to shed more light on the likelihood of false accepts. For example, using a dataset covering a user's daily habits, an algorithm could be devised to assess the security of a trajectory by searching for similar trajectories that occur when the device is *not* in proximity to or co-present with the interacting party. Through a more individualised dataset, it might also be possible to find shorter trajectories that are already characteristic for a given user and location.

- **Additional formation operations:** The idea of vehicle platoons offers improvement in terms of efficiency and safety at its core. However, throughout

this thesis we argued that location information, essential for their implementation, cannot simply be trusted when relying on an external signal source or a third party entity. Yet, when devising security measures it is important to seamlessly integrate with current solutions to not inhibit innovation. Hence, additional protocols to support operations such as platoon splitting and merging without requiring to re-establish the trust between following vehicles and leader vehicles would be beneficial.

- **Physical layer tracking:** In Chapter 6 we provided an outlook as to how the availability of cheap SDRs could be used as a tool to attack cryptography based mix-zone schemes. While approaches such as wireless device fingerprinting have been researched for quite some time, we would argue that they are only applicable for targeted attacks since a model has to be trained for each individual OBU. To assess the full impact of SDRs, more research in location based features of the physical layer properties during wireless signal propagation is needed. If car transmitters could be tracked without vehicle specific training, this would open the door for surveillance at an unprecedented scale. This becomes particularly important since VANET standards are close to being implemented in customer products.

- **Location information reliability:** Developers of smart phone applications can obtain their user's location information though a few library imports and by adding a permission request to their project manifest. Similarly, by adding a GPS module to an OBU, location data can be easily made available. However, in both cases it is necessary to be aware that this information is not fully reliable. While a banking app might use it as a countermeasure for fraud, which is usually covered by a bank's insurance, an OBU transmits this information in singed CAMs to other vehicles. One important question that arises is: *Who is responsible if an accident happens due to false location information?* One could argue that the signing vehicle should have verified the location information before propagating it and thus is responsible. On

the other hand, one could say that every vehicle should treat an external data source as untrustworthy and the receiving vehicle should have verified the claim. Research on the policy and the legal implications of location information has to take these considerations into account and find a way to handle such cases.

- **Real-world VANET deployments:** Although significant efforts were put into the creation of simulation tools like SUMO, Omnet++, and the Veins framework, extensive tests on real-world deployments are still lagging behind. Some sites in the North of France, the Netherlands, and the city of Porto in Portugal, have been equipped with RSUs for initial field trials. While these mostly focus on important challenges in signalling technology, such as antenna radiation patterns, their placement, and communication characteristics, research on physical property verification and passenger privacy could greatly benefit from the datasets produced by these projects. Ideally, the data would be used to validate the simulation results obtained so far.

8.3 Final Conclusions

The segregation of the digital and physical world was for decades a given fact in computer security. Throughout recent years, this lead to the development of a wide range of cryptographic methods allowing the attribution of digital properties to pieces of information. However, with the advent of cyber-physical systems, such as smart door locks, mobile payment terminals, and modern vehicles, the line between the digital and the physical world becomes increasingly blurry. Hence, when providing digitally controlled systems with the capability to manipulate physical objects, it is necessary to verify not just digital but also physical properties.

Moreover, with the advent of the internet of everything, we are just at the beginning of a deep integration of digital technology into our daily lives. Cyber-physical system will require newly developed mechanisms that allow for the certification of physical properties in particular when involved in life-critical decisions. In the

future, autonomous vehicles and unmanned drones are going to encounter humans in the course of their duty as we have already seen in recent field trials. Despite advances in sensing technology, such as LIDAR, it is necessary to also consider adversarial input either presented to the sensors themselves or introduced through the data received from a third party. As such, similar to the safety requirements once imposed on car makers, security and privacy standards should play a primary role in the development process of any cyber-physical product.

Appendices

A

Turning Angle Error

This appendix provides the turning angle error between two smart phones of different make and model placed at separate positions in the experimental vehicle.

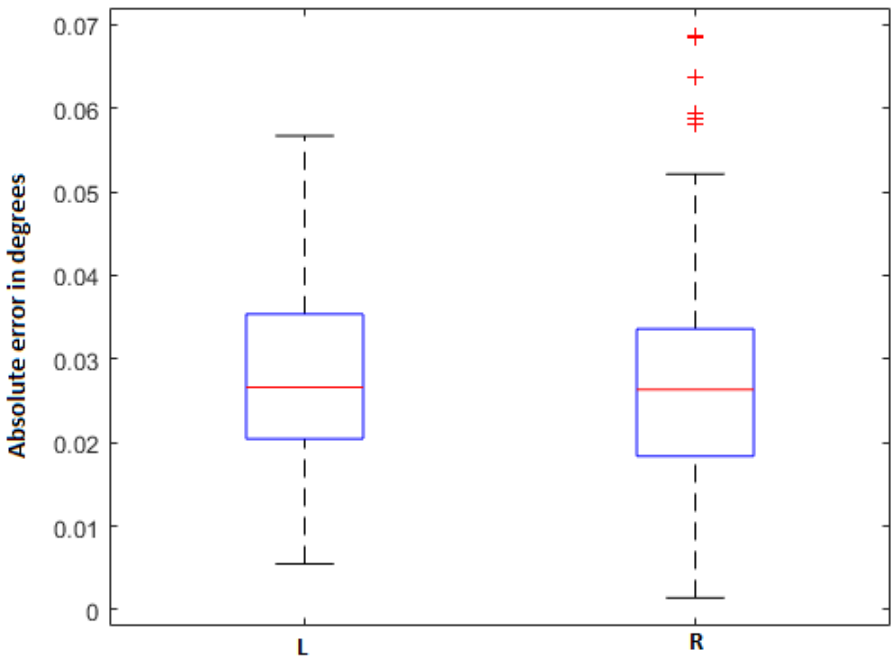


Figure A.1: Absolute measurement error of angles for left an right turns captured at 20 Hz using two phones placed in the experimental vehicle.

References

[1] Bernhard Hofmann-Wellenhof, Klaus Legat, and Manfred Wieser. *Navigation: principles of positioning and guidance*. Springer Science & Business Media, 2011.

[2] Otto Neugebauer. "The Early History of the Astrolabe. Studies in Ancient Astronomy IX". In: *Isis* 40.3 (1949), pp. 240–256.

[3] Frederic C Lane. "The economic meaning of the invention of the compass". In: *The American Historical Review* 68.3 (1963), pp. 605–617.

[4] Bernhard Hofmann-Wellenhof, Herbert Lichtenegger, and Elmar Wasle. *GNSS–global navigation satellite systems: GPS, GLONASS, Galileo, and more*. Springer Science & Business Media, 2007.

[5] Joseph Bonneau et al. "Passwords and the evolution of imperfect authentication". In: *Commun. ACM* 58.7 (2015), pp. 78–87. URL: http://doi.acm.org/10.1145/2699390.

[6] Grady Xiao, Mariofanna Milanova, and Mengjun Xie. "Secure behavioral biometric authentication with leap motion". In: *2016 4th International Symposium on Digital Forensic and Security (ISDFS)*. IEEE, 2016.

[7] Anastasios Tefas, Constantine Kotropoulos, and Ioannis Pitas. "Using Support Vector Machines to Enhance the Performance of Elastic Graph Matching for Frontal Face Authentication". In: *IEEE Trans. Pattern Anal. Mach. Intell.* 23.7 (2001), pp. 735–746. URL: https://doi.org/10.1109/34.935847.

[8] Hien Thi Thu Truong et al. "Comparing and fusing different sensor modalities for relay attack resistance in Zero-Interaction Authentication". In: *IEEE International Conference on Pervasive Computing and Communications, PerCom 2014, Budapest, Hungary, March 24-28, 2014*. 2014, pp. 163–171. URL: https://doi.org/10.1109/PerCom.2014.6813957.

[9] Mark D. Corner and Brian Noble. "Zero-interaction authentication". In: *Proceedings of the Eighth Annual International Conference on Mobile Computing and Networking, MOBICOM 2002, Atlanta, Georgia, USA, September 23-28, 2002*. Ed. by Ian F. Akyildiz et al. ACM, 2002, pp. 1–11. URL: https://doi.org/10.1145/570645.570647.

[10] SourceForge. *BlueProximity*. https://sourceforge.net/projects/blueproximity/.

[11] Aurélien Francillon, Boris Danev, and Srdjan Capkun. "Relay Attacks on Passive Keyless Entry and Start Systems in Modern Cars". In: *Proceedings of the Network and Distributed System Security Symposium, NDSS 2011, San Diego, California, USA, 6th February - 9th February 2011*. The Internet Society, 2011. URL: http://www.isoc.org/isoc/conferences/ndss/11/pdf/2%5C_1.pdf.

[12] Flavio D. Garcia et al. “Lock It and Still Lose It - on the (In)Security of Automotive Remote Keyless Entry Systems”. In: *25th USENIX Security Symposium, USENIX Security 16, Austin, TX, USA, August 10-12, 2016*. Ed. by Thorsten Holz and Stefan Savage. USENIX Association, 2016. URL: https://www.usenix.org/conference/usenixsecurity16/technical-sessions/presentation/garcia.

[13] Lishoy Francis et al. “Practical Relay Attack on Contactless Transactions by Using NFC Mobile Phones”. In: *IACR Cryptology ePrint Archive* 2011 (2011), p. 618. URL: http://eprint.iacr.org/2011/618.

[14] Piotr Szczurek et al. “Estimating Relevance for the Emergency Electronic Brake Light Application”. In: *IEEE Trans. Intelligent Transportation Systems* 13.4 (2012), pp. 1638–1656. URL: https://doi.org/10.1109/TITS.2012.2201257.

[15] George K. Mitropoulos et al. “Wireless Local Danger Warning: Cooperative Foresighted Driving Using Intervehicle Communication”. In: *IEEE Trans. Intelligent Transportation Systems* 11.3 (2010), pp. 539–553. URL: https://doi.org/10.1109/TITS.2009.2034839.

[16] Jennie Lioris et al. “Platoons of connected vehicles can double throughput in urban roads”. In: *Transportation Research Part C: Emerging Technologies* 77 (2017), pp. 292–305.

[17] Kuo-Yun Liang, Jonas Mårtensson, and Karl Henrik Johansson. “Heavy-Duty Vehicle Platoon Formation for Fuel Efficiency”. In: *IEEE Trans. Intelligent Transportation Systems* 17.4 (2016), pp. 1051–1061. URL: https://doi.org/10.1109/TITS.2015.2492243.

[18] Bruce DeBruhl et al. “Is your commute driving you crazy?: a study of misbehavior in vehicular platoons”. In: *Proceedings of the 8th ACM Conference on Security & Privacy in Wireless and Mobile Networks, New York, NY, USA, June 22-26, 2015*. ACM, 2015, 22:1–22:11. URL: https://doi.org/10.1145/2766498.2766505.

[19] Norbert Bißmeyer et al. “Central misbehavior evaluation for VANETs based on mobility data plausibility”. In: *Proceedings of the ninth ACM international workshop on Vehicular inter-networking, systems, and applications, VANET ’12, Low Wood Bay, Lake District, UK, June 25, 2012*. Ed. by John B. Kenney et al. ACM, 2012, pp. 73–82. URL: http://doi.acm.org/10.1145/2307888.2307902.

[20] Ahren Studer, Mark Luk, and Adrian Perrig. “Efficient mechanisms to provide convoy member and vehicle sequence authentication in VANETs”. In: *Third International Conference on Security and Privacy in Communication Networks and the Workshops, SecureComm 2007, Nice, France, 17-21 September, 2007*. 2007, pp. 422–432. URL: https://doi.org/10.1109/SECCOM.2007.4550363.

[21] Karim Emara, Wolfgang Woerndl, and Johann H. Schlichter. “Vehicle tracking using vehicular network beacons”. In: *IEEE 14th International Symposium on "A World of Wireless, Mobile and Multimedia Networks", WoWMoM 2013, Madrid, Spain, June 4-7, 2013*. IEEE Computer Society, 2013, pp. 1–6. URL: https://doi.org/10.1109/WoWMoM.2013.6583473.

[22] Baik Hoh et al. “Achieving Guaranteed Anonymity in GPS Traces via Uncertainty-Aware Path Cloaking”. In: *IEEE Trans. Mob. Comput.* 9.8 (2010), pp. 1089–1107. URL: https://doi.org/10.1109/TMC.2010.62.

[23] Sashank Narain et al. “Inferring User Routes and Locations Using Zero-Permission Mobile Sensors”. In: *IEEE Symposium on Security and Privacy, SP 2016, San Jose, CA, USA, May 22-26, 2016*. IEEE Computer Society, 2016, pp. 397–413. URL: https://doi.org/10.1109/SP.2016.31.

[24] Xianyi Gao et al. “Elastic pathing: your speed is enough to track you”. In: *The 2014 ACM Conference on Ubiquitous Computing, UbiComp ’14, Seattle, WA, USA, September 13-17, 2014*. 2014, pp. 975–986. URL: http://doi.acm.org/10.1145/2632048.2632077.

[25] Takuya Watanabe, Mitsuaki Akiyama, and Tatsuya Mori. “RouteDetector: Sensor-based Positioning System That Exploits Spatio-Temporal Regularity of Human Mobility”. In: *9th USENIX Workshop on Offensive Technologies, WOOT ’15, Washington, DC, USA, August 10-11, 2015.* Ed. by Aurélien Francillon and Thomas Ptacek. USENIX Association, 2015. URL: https://www.usenix.org/conference/woot15/workshop-program/presentation/watanabe.

[26] Z. Berman and J. D. Powell. “The role of dead reckoning and inertial sensors in future general aviation navigation”. In: *IEEE 1998 Position Location and Navigation Symposium (Cat. No.98CH36153)*. Apr. 1998, pp. 510–517. URL: https://doi.org/10.1109/PLANS.1998.670206.

[27] Bo Wei, Niki Trigoni, and Andrew Markham. “iMag: Accurate and Rapidly Deployable Inertial Magneto-Inductive Localisation”. In: *2018 IEEE International Conference on Robotics and Automation, ICRA 2018, Brisbane, Australia, May 21-25, 2018*. IEEE, 2018, pp. 99–106. URL: https://doi.org/10.1109/ICRA.2018.8460804.

[28] A. R. Jimenez et al. “A comparison of Pedestrian Dead-Reckoning algorithms using a low-cost MEMS IMU”. In: *2009 IEEE International Symposium on Intelligent Signal Processing*. Aug. 2009, pp. 37–42.

[29] Nils Ole Tippenhauer et al. “On the requirements for successful GPS spoofing attacks”. In: *Proceedings of the 18th ACM Conference on Computer and Communications Security, CCS 2011, Chicago, Illinois, USA, October 17-21, 2011*. Ed. by Yan Chen, George Danezis, and Vitaly Shmatikov. ACM, 2011, pp. 75–86. URL: http://doi.acm.org/10.1145/2046707.2046719.

[30] Min Choi et al. “Location based authentication scheme using BLE for high performance digital content management system”. In: *Neurocomputing* 209 (2016), pp. 25–38. URL: https://doi.org/10.1016/j.neucom.2015.05.142.

[31] Alexei Czeskis et al. “Strengthening user authentication through opportunistic cryptographic identity assertions”. In: *the ACM Conference on Computer and Communications Security, CCS’12, Raleigh, NC, USA, October 16-18, 2012*. Ed. by Ting Yu, George Danezis, and Virgil D. Gligor. ACM, 2012, pp. 404–414. URL: http://doi.acm.org/10.1145/2382196.2382240.

[32] Kulpreet Chilana and Federico Casalegno. “Hermes: Hands-Free Authentication in Physical Spaces”. In: *Human Aspects of Information Security, Privacy, and Trust - 4th International Conference, HAS 2016, Held as Part of HCI International 2016, Toronto, ON, Canada, July 17-22, 2016, Proceedings*. 2016, pp. 15–24. URL: https://doi.org/10.1007/978-3-319-39381-0%5C_2.

[33] Danny Dolev and Andrew Chi-Chih Yao. "On the security of public key protocols". In: *IEEE Trans. Information Theory* 29.2 (1983), pp. 198–207. URL: https://doi.org/10.1109/TIT.1983.1056650.

[34] Albert Levi et al. "Relay Attacks on Bluetooth Authentication and Solutions". In: *Computer and Information Sciences - ISCIS 2004, 19th International Symposium, Kemer-Antalya, Turkey, October 27-29, 2004. Proceedings*. Ed. by Cevdet Aykanat, Tugrul Dayar, and Ibrahim Korpeoglu. Vol. 3280. Lecture Notes in Computer Science. Springer, 2004, pp. 278–288. URL: https://doi.org/10.1007/978-3-540-30182-0%5C_29.

[35] Ziv Kfir and Avishai Wool. "Picking Virtual Pockets using Relay Attacks on Contactless Smartcard Systems". In: *IACR Cryptology ePrint Archive* 2005 (2005), p. 52. URL: http://eprint.iacr.org/2005/052.

[36] Luigi Sportiello and Andrea Ciardulli. "Long Distance Relay Attack". In: *Radio Frequency Identification - Security and Privacy Issues 9th International Workshop, RFIDsec 2013, Graz, Austria, July 9-11, 2013, Revised Selected Papers*. Ed. by Michael Hutter and Jörn-Marc Schmidt. Vol. 8262. Lecture Notes in Computer Science. Springer, 2013, pp. 69–85. URL: https://doi.org/10.1007/978-3-642-41332-2%5C_5.

[37] Daniel P Shepard et al. "Evaluation of smart grid and civilian UAV vulnerability to GPS spoofing attacks". In: *Radionavigation Laboratory Conference Proceedings*. 2012.

[38] S. M. Giray. "Anatomy of unmanned aerial vehicle hijacking with signal spoofing". In: *2013 6th International Conference on Recent Advances in Space Technologies (RAST)*. June 2013, pp. 795–800. URL: https://doi.org/10.1109/RAST.2013.6581320.

[39] Andrew J. Kerns et al. "Unmanned Aircraft Capture and Control Via GPS Spoofing". In: *J. Field Robotics* 31.4 (2014), pp. 617–636. URL: https://doi.org/10.1002/rob.21513.

[40] Edwin Vattapparamban et al. "Drones for smart cities: Issues in cybersecurity, privacy, and public safety". In: *2016 International Wireless Communications and Mobile Computing Conference (IWCMC), Paphos, Cyprus, September 5-9, 2016*. Ed. by Mario Gerla, George C. Hadjichristofi, and Mohsen Guizani. IEEE, 2016, pp. 216–221. URL: https://doi.org/10.1109/IWCMC.2016.7577060.

[41] Philippe Golle, Daniel H. Greene, and Jessica Staddon. "Detecting and correcting malicious data in VANETs". In: *Proceedings of the First International Workshop on Vehicular Ad Hoc Networks, 2004, Philadelphia, PA, USA, October 1, 2004*. Ed. by Kenneth P. Laberteaux et al. ACM, 2004, pp. 29–37. URL: https://doi.org/10.1145/1023875.1023881.

[42] Stefan Brands and David Chaum. "Distance-Bounding Protocols (Extended Abstract)". In: *Advances in Cryptology - EUROCRYPT '93, Workshop on the Theory and Application of of Cryptographic Techniques, Lofthus, Norway, May 23-27, 1993, Proceedings*. 1993, pp. 344–359. URL: https://doi.org/10.1007/3-540-48285-7%5C_30.

[43] Nils Ole Tippenhauer and Srdjan Capkun. "ID-Based Secure Distance Bounding and Localization". In: *Computer Security - ESORICS 2009, 14th European Symposium on Research in Computer Security, Saint-Malo, France, September 21-23, 2009. Proceedings*. Ed. by Michael Backes and Peng Ning. Vol. 5789. Lecture Notes in Computer Science. Springer, 2009, pp. 621–636. URL: https://doi.org/10.1007/978-3-642-04444-1%5C_38.

[44] Kasper Bonne Rasmussen and Srdjan Capkun. "Realization of RF Distance Bounding". In: *19th USENIX Security Symposium, Washington, DC, USA, August 11-13, 2010, Proceedings*. USENIX Association, 2010, pp. 389–402. URL: http://www.usenix.org/events/sec10/tech/full%5C_papers/Rasmussen.pdf.

[45] Aanjhan Ranganathan et al. "Design and Implementation of a Terrorist Fraud Resilient Distance Bounding System". In: *Computer Security - ESORICS 2012 - 17th European Symposium on Research in Computer Security, Pisa, Italy, September 10-12, 2012. Proceedings*. Ed. by Sara Foresti, Moti Yung, and Fabio Martinelli. Vol. 7459. Lecture Notes in Computer Science. Springer, 2012, pp. 415–432. URL: https://doi.org/10.1007/978-3-642-33167-1%5C_24.

[46] Jason Reid et al. "Detecting relay attacks with timing-based protocols". In: *Proceedings of the 2007 ACM Symposium on Information, Computer and Communications Security, ASIACCS 2007, Singapore, March 20-22, 2007*. 2007, pp. 204–213. URL: https://doi.org/10.1145/1229285.1229314.

[47] Gerhard P. Hancke and Markus G. Kuhn. "An RFID Distance Bounding Protocol". In: *First International Conference on Security and Privacy for Emerging Areas in Communications Networks, SecureComm 2005, Athens, Greece, 5-9 September, 2005*. 2005, pp. 67–73. URL: https://doi.org/10.1109/SECURECOMM.2005.56.

[48] Saar Drimer and Steven J. Murdoch. "Keep Your Enemies Close: Distance Bounding Against Smartcard Relay Attacks". In: *Proceedings of the 16th USENIX Security Symposium, Boston, MA, USA, August 6-10, 2007*. 2007. URL: https://www.usenix.org/conference/16th-usenix-security-symposium/keep-your-enemies-close-distance-bounding-against.

[49] Joo-Han Song, Vincent W. S. Wong, and Victor C. M. Leung. "Secure Location Verification for Vehicular Ad-Hoc Networks". In: *Proceedings of the Global Communications Conference, 2008. GLOBECOM 2008, New Orleans, LA, USA, 30 November - 4 December 2008*. IEEE, 2008, pp. 806–810. URL: https://doi.org/10.1109/GLOCOM.2008.ECP.160.

[50] Markus Miettinen et al. "Context-Based Zero-Interaction Pairing and Key Evolution for Advanced Personal Devices". In: *Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security, Scottsdale, AZ, USA, November 3-7, 2014*. 2014, pp. 880–891. URL: https://doi.org/10.1145/2660267.2660334.

[51] Jun Han et al. "Convoy: Physical Context Verification for Vehicle Platoon Admission". In: *Proceedings of the 18th International Workshop on Mobile Computing Systems and Applications, HotMobile 2017, Sonoma, CA, USA, February 21 - 22, 2017*. 2017, pp. 73–78. URL: https://doi.org/10.1145/3032970.3032987.

[52] Tzipora Halevi et al. “Secure Proximity Detection for NFC Devices Based on Ambient Sensor Data”. In: *Computer Security - ESORICS 2012 - 17th European Symposium on Research in Computer Security, Pisa, Italy, September 10-12, 2012. Proceedings*. Ed. by Sara Foresti, Moti Yung, and Fabio Martinelli. Vol. 7459. Lecture Notes in Computer Science. Springer, 2012, pp. 379–396. URL: https://doi.org/10.1007/978-3-642-33167-1%5C_22.

[53] Babins Shrestha et al. “Drone to the Rescue: Relay-Resilient Authentication using Ambient Multi-sensing”. In: *Financial Cryptography and Data Security - 18th International Conference, FC 2014, Christ Church, Barbados, March 3-7, 2014, Revised Selected Papers*. Ed. by Nicolas Christin and Reihaneh Safavi-Naini. Vol. 8437. Lecture Notes in Computer Science. Springer, 2014, pp. 349–364. URL: https://doi.org/10.1007/978-3-662-45472-5%5C_23.

[54] Babins Shrestha et al. “Contextual Proximity Detection in the Face of Context-Manipulating Adversaries”. In: *CoRR* abs/1511.00905 (2015). arXiv: 1511.00905. URL: http://arxiv.org/abs/1511.00905.

[55] Nikolaos Karapanos et al. “Sound-Proof: Usable Two-Factor Authentication Based on Ambient Sound”. In: *24th USENIX Security Symposium, USENIX Security 15, Washington, D.C., USA, August 12-14, 2015*. 2015, pp. 483–498. URL: https://www.usenix.org/conference/usenixsecurity15/technical-sessions/presentation/karapanos.

[56] Sarfraz Nawaz and Cecilia Mascolo. “Mining users’ significant driving routes with low-power sensors”. In: *Proceedings of the 12th ACM Conference on Embedded Network Sensor Systems, SenSys ’14, Memphis, Tennessee, USA, November 3-6, 2014*. Ed. by Ákos Lédeczi, Prabal Dutta, and Chenyang Lu. ACM, 2014, pp. 236–250. URL: http://doi.acm.org/10.1145/2668332.2668348.

[57] Mika Juuti et al. “STASH: Securing Transparent Authentication Schemes Using Prover-Side Proximity Verification”. In: *14th Annual IEEE International Conference on Sensing, Communication, and Networking, SECON 2017, San Diego, CA, USA, June 12-14, 2017*. IEEE, 2017, pp. 1–9. URL: https://doi.org/10.1109/SAHCN.2017.7964922.

[58] Mika Juuti et al. “Implementing Prover-Side Proximity Verification for Strengthening Transparent Authentication”. In: *14th Annual IEEE International Conference on Sensing, Communication, and Networking, SECON 2017, San Diego, CA, USA, June 12-14, 2017*. IEEE, 2017, pp. 1–2. URL: https://doi.org/10.1109/SAHCN.2017.7964951.

[59] Fernando J Corbató et al. *The Compatible Time-Sharing System: A Programmer’s Guide*. Vol. 45. Mit Press Cambridge, Mass., 1963.

[60] Kresimir Delac and Mislav Grgic. “A survey of biometric recognition methods”. In: *46th International Symposium Electronics in Marine*. Vol. 46. 2004, pp. 16–18.

[61] Thanasis Petsas et al. “Two-factor authentication: is the world ready?: quantifying 2FA adoption”. In: *Proceedings of the Eighth European Workshop on System Security, EuroSec 2015, Bordeaux, France, April 21, 2015*. Ed. by Juan Caballero and Michalis Polychronakis. ACM, 2015, 4:1–4:7. URL: http://doi.acm.org/10.1145/2751323.2751327.

[62] Vishal M. Patel et al. "Continuous User Authentication on Mobile Devices: Recent progress and remaining challenges". In: *IEEE Signal Process. Mag.* 33.4 (2016), pp. 49–61. URL: https://doi.org/10.1109/MSP.2016.2555335.

[63] Navid Nourani-Vatani, Jonathan M. Roberts, and Mandyam V. Srinivasan. "Practical visual odometry for car-like vehicles". In: *2009 IEEE International Conference on Robotics and Automation, ICRA 2009, Kobe, Japan, May 12-17, 2009*. 2009, pp. 3551–3557. URL: https://doi.org/10.1109/ROBOT.2009.5152403.

[64] Oliver J Woodman. "An introduction to inertial navigation". In: *University of Cambridge, Computer Laboratory, Tech. Rep. UCAMCL-TR-696* 14 (2007), p. 15.

[65] Kevin P. Murphy. *Machine learning - a probabilistic perspective*. Adaptive computation and machine learning series. MIT Press, 2012.

[66] Richard Durbin et al. *Biological Sequence Analysis: Probabilistic Models of Proteins and Nucleic Acids*. Cambridge University Press, 1998.

[67] Eric Brill and Robert C. Moore. "An Improved Error Model for Noisy Channel Spelling Correction". In: *38th Annual Meeting of the Association for Computational Linguistics, Hong Kong, China, October 1-8, 2000*. ACL, 2000. URL: http://www.aclweb.org/anthology/P00-1037.

[68] Mark A. Hall et al. "The WEKA data mining software: an update". In: *SIGKDD Explorations* 11.1 (2009), pp. 10–18. URL: http://doi.acm.org/10.1145/1656274.1656278.

[69] Android. *Keeping the device awake*. https://developer.android.com/training/scheduling/wakelock.html. 2016.

[70] Luis Corral et al. "A method for characterizing energy consumption in Android smartphones". In: *2nd International Workshop on Green and Sustainable Software, GREENS 2013, San Francisco, CA, USA, May 20, 2013*. 2013, pp. 38–45. URL: https://doi.org/10.1109/GREENS.2013.6606420.

[71] Ella Peltonen et al. "Energy modeling of system settings: A crowdsourced approach". In: *2015 IEEE International Conference on Pervasive Computing and Communications, PerCom 2015, St. Louis, MO, USA, 23-27 March, 2015*. IEEE Computer Society, 2015, pp. 37–45. URL: https://doi.org/10.1109/PERCOM.2015.7146507.

[72] Alex Hern. *Google aims to kill passwords by the end of this year | Technology | The Guardian*. Aug. 2016. URL: https://www.theguardian.com/technology/2016/may/24/google-passwords-android.

[73] Samuli Hemminki, Petteri Nurmi, and Sasu Tarkoma. "Accelerometer-based transportation mode detection on smartphones". In: *The 11th ACM Conference on Embedded Network Sensor Systems, SenSys '13, Roma, Italy, November 11-15, 2013*. Ed. by Chiara Petrioli, Landon P. Cox, and Kamin Whitehouse. ACM, 2013, 13:1–13:14. URL: https://doi.org/10.1145/2517351.2517367.

[74] National Highway Traffic Safety Administration (NHTSA). "Traffic Safety Facts: 2015". In: *U.S. Department of Transportation* August (2015), pp. 1–9.

[75] Meredith Pedde, Adam A Szpiro, and Sara D Adar. "Traffic congestion as a risk factor for mortality in near-road communities: a case-crossover study". In: *American journal of epidemiology* 186.5 (2017), pp. 564–572.

[76] Rui Ding et al. "Characteristics of DNA methylation changes induced by traffic-related air pollution". In: *Mutation Research/Genetic Toxicology and Environmental Mutagenesis* 796 (2016), pp. 46–53.

[77] Nicola George and Kathryn Kershaw. "Road Use Statistics Great Britain". In: *UK Department of Transport* (2016), p. 41.

[78] Evdoxia Tsakiridou. *How Vehicle-to-Vehicle Communication Could Replace Traffic Lights and Shorten Commutes*. Nov. 2018. URL: https://www.fortiss.org/en/about-us/alle-news/safe-connected-driving-on-the-digitally-supported-highway/.

[79] Philip E Ross. "Thus spoke the autobahn". In: *IEEE Spectrum* 52.1 (2015), pp. 52–55.

[80] Arturo Dávila and Mario Nombela. "Sartre: Safe road trains for the environment". In: *Conference on Personal Rapid Transit PRT@ LHR*. Vol. 3. 2010, pp. 2–3.

[81] Tim Leinmüller et al. "Sevecom-secure vehicle communication". In: *IST Mobile and Wireless Communication Summit*. LCA-POSTER-2008-005. 2006.

[82] A. Festag and S. Hess. "ETSI technical committee ITS: news from european standardization for intelligent transport systems (ITS)- [global communications newsletter]". In: *IEEE Communications Magazine* 47.6 (June 2009), pp. 1–4.

[83] Qualcomm Technologies. *Leading the World to 5G: Cellular Vehicle-to-Everything (C-V2X) Technologies*. tiny.cc/u4v2xy. June 2016.

[84] Daniel Jiang and Luca Delgrossi. "IEEE 802.11p: Towards an International Standard for Wireless Access in Vehicular Environments". In: *Proceedings of the 67th IEEE Vehicular Technology Conference, VTC Spring 2008, 11-14 May 2008, Singapore*. IEEE, 2008, pp. 2036–2040. URL: https://doi.org/10.1109/VETECS.2008.458.

[85] ETSI. "Intelligent Transport Systems (ITS); Security; Security Header and Certificate Formats". In: *EN TS 103 097 v1.1.1* 1 (2013).

[86] John B. Kenney. "Dedicated Short-Range Communications (DSRC) Standards in the United States". In: *Proceedings of the IEEE* 99.7 (2011), pp. 1162–1182. URL: https://doi.org/10.1109/JPROC.2011.2132790.

[87] Yunpeng Zang et al. "Wireless Local Danger Warning Using Inter-Vehicle Communications in Highway Scenarios". In: *European Wireless 2008 - 14th European Wireless Conference 2008, Electronic Proceedings, 22-25 June 2008, Praque, Czech Republic*. VDE-Verlag, 2008. URL: http://www.vde-verlag.de/proceedings-de/563102068.html.

[88] Michael R. Hafner et al. "Cooperative Collision Avoidance at Intersections: Algorithms and Experiments". In: *IEEE Trans. Intelligent Transportation Systems* 14.3 (2013), pp. 1162–1175. URL: https://doi.org/10.1109/TITS.2013.2252901.

[89] James A Misener, Raja Sengupta, and Hariharan Krishnan. "Cooperative collision warning: Enabling crash avoidance with wireless technology". In: *12th World Congress on Intelligent Transport SystemsITS AmericaITS JapanERTICO*. 2005.

[90] Michael P Lammert et al. "Effect of platooning on fuel consumption of class 8 vehicles over a range of speeds, following distances, and mass". In: *SAE International Journal of Commercial Vehicles* 7.2014-01-2438 (2014), pp. 626–639.

[91] Valerio Turri, Bart Besselink, and Karl Henrik Johansson. "Cooperative Look-Ahead Control for Fuel-Efficient and Safe Heavy-Duty Vehicle Platooning". In: *IEEE Trans. Contr. Sys. Techn.* 25.1 (2017), pp. 12–28. URL: https://doi.org/10.1109/TCST.2016.2542044.

[92] Nikita Lyamin, Qichen Dengy, and Alexey V. Vinel. "Study of the platooning fuel efficiency under ETSI ITS-G5 communications". In: *19th IEEE International Conference on Intelligent Transportation Systems, ITSC 2016, Rio de Janeiro, Brazil, November 1-4, 2016*. 2016, pp. 551–556. URL: https://doi.org/10.1109/ITSC.2016.7795608.

[93] Amr Abdelaziz, Ron Burton, and Can Emre Koksal. "Message Authentication and Secret Key Agreement in VANETs via Angle of Arrival". In: *CoRR* abs/1609.03109 (2016). arXiv: 1609.03109. URL: http://arxiv.org/abs/1609.03109.

[94] Alan Ali, Gaëtan Garcia, and Philippe Martinet. "Minimizing the inter-vehicle distances of the time headway policy for urban platoon control with decoupled longitudinal and lateral control". In: *16th International IEEE Conference on Intelligent Transportation Systems, ITSC 2013, The Hague, The Netherlands, October 6-9, 2013*. 2013, pp. 1805–1810. URL: https://doi.org/10.1109/ITSC.2013.6728490.

[95] Ozan K. Tonguz. *How Vehicle-to-Vehicle Communication Could Replace Traffic Lights and Shorten Commutes*. Aug. 2018. URL: https://spectrum.ieee.org/transportation/infrastructure/how-vehicletovehicle-communication-could-replace-traffic-lights-and-shorten-commutes.

[96] Panos Papadimitratos et al. "Vehicular communication systems: Enabling technologies, applications, and future outlook on intelligent transportation". In: *IEEE Communications Magazine* 47.11 (2009), pp. 84–95. URL: https://doi.org/10.1109/MCOM.2009.5307471.

[97] Roberto A Uzcátegui, Antonio Jose De Sucre, and Guillermo Acosta-Marum. "Wave: A tutorial". In: *IEEE Communications magazine* 47.5 (2009).

[98] Andreas Buchenscheit et al. "A VANET-based emergency vehicle warning system". In: *IEEE Vehicular Networking Conference, VNC 2009, Tokyo, Japan, October 28-30, 2009*. IEEE, 2009, pp. 1–8. URL: https://doi.org/10.1109/VNC.2009.5416384.

[99] John R. Douceur. "The Sybil Attack". In: *Peer-to-Peer Systems, First International Workshop, IPTPS 2002, Cambridge, MA, USA, March 7-8, 2002, Revised Papers*. Ed. by Peter Druschel, M. Frans Kaashoek, and Antony I. T. Rowstron. Vol. 2429. Lecture Notes in Computer Science. Springer, 2002, pp. 251–260. URL: https://doi.org/10.1007/3-540-45748-8%5C_24.

[100] Hugo Krawczyk, Mihir Bellare, and Ran Canetti. "HMAC: Keyed-Hashing for Message Authentication". In: *RFC* 2104 (1997), pp. 1–11. URL: https://doi.org/10.17487/RFC2104.

[101] Adrian McCullagh and William J. Caelli. "Non-Repudiation in the Digital Environment". In: *First Monday* 5.8 (2000). URL: http://firstmonday.org/htbin/cgiwrap/bin/ojs/index.php/fm/article/view/778.

[102] Lars Fischer et al. "Secure revocable anonymous authenticated inter-vehicle communication (SRAAC)". In: *4th Conference on Embedded Security in Cars (ESCAR 2006), Berlin, Germany*. Citeseer. 2006.

[103] Florian Schaub et al. "V-Tokens for Conditional Pseudonymity in VANETs". In: *2010 IEEE Wireless Communications and Networking Conference, WCNC 2010, Proceedings, Sydney, Australia, 18-21 April 2010*. IEEE, 2010, pp. 1–6. URL: https://doi.org/10.1109/WCNC.2010.5506126.

[104] Zhendong Ma, Frank Kargl, and Michael Weber. "Pseudonym-On-Demand: A New Pseudonym Refill Strategy for Vehicular Communications". In: *Proceedings of the 68th IEEE Vehicular Technology Conference, VTC Fall 2008, 21-24 September 2008, Calgary, Alberta, Canada*. IEEE, 2008, pp. 1–5. URL: https://doi.org/10.1109/VETECF.2008.455.

[105] Mohammad Khodaei and Panos Papadimitratos. "Efficient, Scalable, and Resilient Vehicle-Centric Certificate Revocation List Distribution in VANETs". In: *Proceedings of the 11th ACM Conference on Security & Privacy in Wireless and Mobile Networks, WiSec 2018, Stockholm, Sweden, June 18-20, 2018*. Ed. by Panos Papadimitratos, Kevin Butler, and Christina Pöpper. ACM, 2018, pp. 172–183. URL: http://doi.acm.org/10.1145/3212480.3212481.

[106] Shushan Zhao et al. "A Survey of Applications of Identity-Based Cryptography in Mobile Ad-Hoc Networks". In: *IEEE Communications Surveys and Tutorials* 14.2 (2012), pp. 380–400. URL: https://doi.org/10.1109/SURV.2011.020211.00045.

[107] Adi Shamir. "Identity-Based Cryptosystems and Signature Schemes". In: *Advances in Cryptology, Proceedings of CRYPTO '84, Santa Barbara, California, USA, August 19-22, 1984, Proceedings*. Ed. by G. R. Blakley and David Chaum. Vol. 196. Lecture Notes in Computer Science. Springer, 1984, pp. 47–53. URL: https://doi.org/10.1007/3-540-39568-7%5C_5.

[108] Dan Boneh and Matthew K. Franklin. "Identity-Based Encryption from the Weil Pairing". In: *Advances in Cryptology - CRYPTO 2001, 21st Annual International Cryptology Conference, Santa Barbara, California, USA, August 19-23, 2001, Proceedings*. Ed. by Joe Kilian. Vol. 2139. Lecture Notes in Computer Science. Springer, 2001, pp. 213–229. URL: https://doi.org/10.1007/3-540-44647-8%5C_13.

[109] Jianhong Zhang and Yuwei Xu. "Breaking and Repairing of an Anonymous and Traceable Communication Protocol for Vehicular Ad Hoc Networks". In: *12th IEEE International Conference on Computer and Information Technology, CIT 2012, Chengdu, Sichuan, China, October 27-29, 2012*. IEEE Computer Society, 2012, pp. 88–93. URL: https://doi.org/10.1109/CIT.2012.41.

[110] Kenneth G. Paterson and Geraint Price. "A comparison between traditional public key infrastructures and identity-based cryptography". In: *Inf. Sec. Techn. Report* 8.3 (2003), pp. 57–72. URL: https://doi.org/10.1016/S1363-4127(03)00308-X.

[111] Giorgio Calandriello et al. "Efficient and robust pseudonymous authentication in VANET". In: *Proceedings of the Fourth International Workshop on Vehicular Ad Hoc Networks, VANET 2007, Montréal, Québec, Canada, September 10, 2007*. Ed. by Wieland Holfelder et al. ACM, 2007, pp. 19–28. URL: http://doi.acm.org/10.1145/1287748.1287752.

[112] Jinhua Guo, John P Baugh, and Shengquan Wang. "A group signature based secure and privacy-preserving vehicular communication framework". In: *2007 Mobile Networking for Vehicular Environments*. IEEE. 2007, pp. 103–108.

[113] Chenxi Zhang et al. "An Efficient Message Authentication Scheme for Vehicular Communications". In: *IEEE Trans. Vehicular Technology* 57.6 (2008), pp. 3357–3368. URL: https://doi.org/10.1109/TVT.2008.928581.

[114] Jong Youl Choi, Markus Jakobsson, and Susanne Wetzel. "Balancing auditability and privacy in vehicular networks". In: *Q2SWinet'05 - Proceedings of the First ACM Workshop on Q2S and Security for Wireless and Mobile Networks, Montreal, Quebec, Canada, October 13, 2005*. Ed. by Azzedine Boukerche and Regina Borges de Araujo. ACM, 2005, pp. 79–87. URL: http://doi.acm.org/10.1145/1089761.1089775.

[115] Chenxi Zhang et al. "RAISE: An Efficient RSU-Aided Message Authentication Scheme in Vehicular Communication Networks". In: *Proceedings of IEEE International Conference on Communications, ICC 2008, Beijing, China, 19-23 May 2008*. IEEE, 2008, pp. 1451–1457. URL: https://doi.org/10.1109/ICC.2008.281.

[116] Adrian Perrig et al. "The TESLA broadcast authentication protocol". In: *Rsa Cryptobytes* 5 (2005).

[117] Jonathan Petit et al. "Pseudonym Schemes in Vehicular Networks: A Survey". In: *IEEE Communications Surveys and Tutorials* 17.1 (2015), pp. 228–255. URL: https://doi.org/10.1109/COMST.2014.2345420.

[118] Magda El Zarki et al. "Security issues in a future vehicular network". In: *European Wireless*. Vol. 2. 2002.

[119] "IEEE Standard for Wireless Access in Vehicular Environments (WAVE) –Networking Services". In: *IEEE Vehicular Technology Society* (Jan. 2016).

[120] ETSI. "Intelligent Transport Systems (ITS); Vehicular Communications; Basic Set of Applications; Part 1 to 3". In: *EN 302 637-1-3* 1 (2010).

[121] Maxim Raya, Adel Aziz, and Jean-Pierre Hubaux. "Efficient secure aggregation in VANETs". In: *Proceedings of the Third International Workshop on Vehicular Ad Hoc Networks, VANET 2006, Los Angeles, CA, USA, September 29, 2007*. Ed. by Wieland Holfelder et al. ACM, 2006, pp. 67–75. URL: http://doi.acm.org/10.1145/1161064.1161076.

[122] Mohammad Khodaei, Hongyu Jin, and Panos Papadimitratos. "Towards deploying a scalable & robust vehicular identity and credential management infrastructure". In: *2014 IEEE Vehicular Networking Conference, VNC 2014, Paderborn, Germany, December 3-5, 2014*. IEEE, 2014, pp. 33–40. URL: https://doi.org/10.1109/VNC.2014.7013306.

[123] Mohammad Khodaei, Hongyu Jin, and Panagiotis Papadimitratos. "SECMACE: Scalable and Robust Identity and Credential Management Infrastructure in Vehicular Communication Systems". In: *IEEE Trans. Intelligent Transportation Systems* 19.5 (2018), pp. 1430–1444. URL: https://doi.org/10.1109/TITS.2017.2722688.

[124] Mohammad Khodaei and Panos Papadimitratos. "The key to intelligent transportation: Identity and credential management in vehicular communication systems". In: *IEEE Vehicular Technology Magazine* 10.4 (2015), pp. 63–69.

[125] Mohammad Khodaei and Panos Papadimitratos. "Evaluating on-demand pseudonym acquisition policies in vehicular communication systems". In: *Proceedings of the First International Workshop on Internet of Vehicles and Vehicles of Internet, IoV-VoI '16, Paderborn, Germany, July 4-8, 2016*. Ed. by Mario Gerla et al. ACM, 2016, pp. 7–12. URL: http://doi.acm.org/10.1145/2938681.2938684.

[126] Jacob Bellatti et al. "Driving Habits Data: Location Privacy Implications and Solutions". In: *IEEE Security & Privacy* 15.1 (2017), pp. 12–20. URL: https://doi.org/10.1109/MSP.2017.6.

[127] Björn Wiedersheim et al. "Privacy in inter-vehicular networks: Why simple pseudonym change is not enough". In: *Wireless On-demand Network Systems and Services (WONS), 2010 Seventh International Conference on*. IEEE. 2010, pp. 176–183.

[128] Krishna Sampigethaya et al. "AMOEBA: Robust Location Privacy Scheme for VANET". In: *IEEE Journal on Selected Areas in Communications* 25.8 (2007), pp. 1569–1589. URL: https://doi.org/10.1109/JSAC.2007.071007.

[129] David Förster et al. "An Evaluation of Pseudonym Changes for Vehicular Networks in Large-Scale, Realistic Traffic Scenarios". In: *IEEE Transactions on Intelligent Transportation Systems* (2017).

[130] Matthias Schäfer et al. "Bringing up OpenSky: a large-scale ADS-B sensor network for research". In: *IPSN'14, Proceedings of the 13th International Symposium on Information Processing in Sensor Networks (part of CPS Week), April 15-17, 2014, Berlin, Germany*. Ed. by Adam Wolisz, Jie Liu, and Lin Zhong. IEEE/ACM, 2014, pp. 83–94. URL: http://dl.acm.org/citation.cfm?id=2602350.

[131] Martin Strohmeier. "Security in next generation air traffic communication networks". PhD thesis. University of Oxford, UK, 2016. URL: http://ethos.bl.uk/OrderDetails.do?uin=uk.bl.ethos.730231.

[132] Panagiotis Papadimitratos, Virgil Gligor, and J-P Hubaux. "Securing vehicular communications-assumptions, requirements, and principles". In: *ESCAR* (2006).

[133] Panagiotis Papadimitratos et al. “Secure vehicular communication systems: design and architecture”. In: *IEEE Communications Magazine* 46.11 (2008), pp. 100–109. URL: https://doi.org/10.1109/MCOM.2008.4689252.

[134] Matthias Gerlach. “Assessing and improving privacy in VANETs”. In: *Workshop on Embedded Security in Cars (ESCAR)* (2006).

[135] Emanuel Fonseca et al. “Support of Anonymity in VANETs - Putting Pseudonymity into Practice”. In: *IEEE Wireless Communications and Networking Conference, WCNC 2007, Hong Kong, China, 11-15 March, 2007*. IEEE, 2007, pp. 3400–3405. URL: https://doi.org/10.1109/WCNC.2007.625.

[136] Levente Buttyán et al. “SLOW: A Practical pseudonym changing scheme for location privacy in VANETs”. In: *IEEE Vehicular Networking Conference, VNC 2009, Tokyo, Japan, October 28-30, 2009*. IEEE, 2009, pp. 1–8. URL: https://doi.org/10.1109/VNC.2009.5416380.

[137] Mohammad Khodaei, Hamid Noroozi, and Panos Papadimitratos. “Privacy Preservation through Uniformity”. In: *Proceedings of the 11th ACM Conference on Security & Privacy in Wireless and Mobile Networks, WiSec 2018, Stockholm, Sweden, June 18-20, 2018*. Ed. by Panos Papadimitratos, Kevin Butler, and Christina Pöpper. ACM, 2018, pp. 279–280. URL: http://doi.acm.org/10.1145/3212480.3226101.

[138] David Eckhoff et al. “SlotSwap: strong and affordable location privacy in intelligent transportation systems”. In: *IEEE Communications Magazine* 49.11 (2011), pp. 126–133. URL: https://doi.org/10.1109/MCOM.2011.6069719.

[139] Matthias Gerlach and Felix Güttler. “Privacy in VANETs using Changing Pseudonyms - Ideal and Real”. In: *Proceedings of the 65th IEEE Vehicular Technology Conference, VTC Spring 2007, 22-25 April 2007, Dublin, Ireland*. IEEE, 2007, pp. 2521–2525. URL: https://doi.org/10.1109/VETECS.2007.519.

[140] Karim Emara. “Poster: PREXT: Privacy extension for Veins VANET simulator”. In: *2016 IEEE Vehicular Networking Conference, VNC 2016, Columbus, OH, USA, December 8-10, 2016*. IEEE, 2016, pp. 1–2. URL: https://doi.org/10.1109/VNC.2016.7835979.

[141] Philippe Golle and Kurt Partridge. “On the Anonymity of Home/Work Location Pairs”. In: *Pervasive Computing, 7th International Conference, Pervasive 2009, Nara, Japan, May 11-14, 2009. Proceedings*. Ed. by Hideyuki Tokuda et al. Vol. 5538. Lecture Notes in Computer Science. Springer, 2009, pp. 390–397. URL: https://doi.org/10.1007/978-3-642-01516-8%5C_26.

[142] Leping Huang et al. “Enhancing wireless location privacy using silent period”. In: *IEEE Wireless Communications and Networking Conference, WCNC 2005, March 13-17, 2005, New Orleans, Louisiana, USA*. IEEE, 2005, pp. 1187–1192. URL: https://doi.org/10.1109/WCNC.2005.1424677.

[143] Krishna Sampigethaya et al. “CARAVAN: Providing location privacy for VANET”. In: *Embedded Security in Cars*. Cologne, Germany, Nov. 2005.

[144] George P. Corser, Alejandro Arenas, and Huirong Fu. "Effect on vehicle safety of nonexistent or silenced basic safety messages". In: *2016 International Conference on Computing, Networking and Communications, ICNC 2016, Kauai, HI, USA, February 15-18, 2016*. IEEE Computer Society, 2016, pp. 1–5. URL: https://doi.org/10.1109/ICCNC.2016.7440550.

[145] Karim Emara, Wolfgang Woerndl, and Johann H. Schlichter. "CAPS: context-aware privacy scheme for VANET safety applications". In: *Proceedings of the 8th ACM Conference on Security & Privacy in Wireless and Mobile Networks, New York, NY, USA, June 22-26, 2015*. ACM, 2015, 21:1–21:12. URL: http://doi.acm.org/10.1145/2766498.2766500.

[146] Julien Freudiger et al. "Mix-zones for location privacy in vehicular networks". In: *ACM Workshop on Wireless Networking for Intelligent Transportation Systems (WiN-ITS)*. LCA-CONF-2007-016. 2007.

[147] Rongxing Lu et al. "Pseudonym Changing at Social Spots: An Effective Strategy for Location Privacy in VANETs". In: *IEEE Trans. Vehicular Technology* 61.1 (2012), pp. 86–96. URL: https://doi.org/10.1109/TVT.2011.2162864.

[148] Andreas Tomandl, Florian Scheuer, and Hannes Federrath. "Simulation-based evaluation of techniques for privacy protection in VANETs". In: *8th IEEE International Conference on Wireless and Mobile Computing, Networking and Communications, WiMob 2012, Barcelona, Spain, October 8-10, 2012*. IEEE Computer Society, 2012, pp. 165–172. URL: https://doi.org/10.1109/WiMOB.2012.6379070.

[149] Albert Wasef and Xuemin (Sherman) Shen. "REP: Location Privacy for VANETs Using Random Encryption Periods". In: *MONET* 15.1 (2010), pp. 172–185. URL: https://doi.org/10.1007/s11036-009-0175-4.

[150] Christian Vaas et al. "Get in Line: Ongoing Co-presence Verification of a Vehicle Formation Based on Driving Trajectories". In: *2018 IEEE European Symposium on Security and Privacy, EuroS&P 2018, London, United Kingdom, April 24-26, 2018*. IEEE, 2018, pp. 199–213. URL: https://doi.org/10.1109/EuroSP.2018.00022.

[151] Jean-Michel Contet et al. "Physics inspired multiagent system for vehicle platooning". In: *6th International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS 2007), Honolulu, Hawaii, USA, May 14-18, 2007*. Ed. by Edmund H. Durfee et al. IFAAMAS, 2007, p. 184. URL: https://doi.org/10.1145/1329125.1329348.

[152] Ji-Zhong Shi and Ji-Ye Zhang. "String stability of infinite-dimension stochastic interconnected large-scale systems with time-varying delay". In: *Int. J. Systems Science* 45.5 (2014), pp. 1035–1041. URL: https://doi.org/10.1080/00207721.2012.744860.

[153] Marcos Eduardo Pivaro Monteiro, João Luiz Rebelatto, and Richard Demo Souza. "Information-Theoretic Location Verification System With Directional Antennas for Vehicular Networks". In: *IEEE Trans. Intelligent Transportation Systems* 17.1 (2016), pp. 93–103. URL: https://doi.org/10.1109/TITS.2015.2460114.

[154] Shihao Yan et al. “Location Verification Systems for VANETs in Rician Fading Channels”. In: *IEEE Trans. Vehicular Technology* 65.7 (2016), pp. 5652–5664. URL: https://doi.org/10.1109/TVT.2015.2453160.

[155] Yang Zhang and Guohong Cao. “V-PADA: Vehicle-Platoon-Aware Data Access in VANETs”. In: *IEEE Trans. Vehicular Technology* 60.5 (2011), pp. 2326–2339. URL: https://doi.org/10.1109/TVT.2011.2148202.

[156] Per A. Hallgren, Claudio Orlandi, and Andrei Sabelfeld. “PrivatePool: Privacy-Preserving Ridesharing”. In: *30th IEEE Computer Security Foundations Symposium, CSF 2017, Santa Barbara, CA, USA, August 21-25, 2017*. IEEE Computer Society, 2017, pp. 276–291. URL: https://doi.org/10.1109/CSF.2017.24.

[157] Mordechai (Muki) Haklay and Patrick Weber. “OpenStreetMap: User-Generated Street Maps”. In: *IEEE Pervasive Computing* 7.4 (2008), pp. 12–18. URL: https://doi.org/10.1109/MPRV.2008.80.

[158] John Krumm. “A survey of computational location privacy”. In: *Personal and Ubiquitous Computing* 13.6 (2009), pp. 391–399. URL: https://doi.org/10.1007/s00779-008-0212-5.

[159] Amina Hossain et al. “On the effectiveness of removing location information from trajectory data for preserving location privacy”. In: *Proceedings of the 9th ACM SIGSPATIAL International Workshop on Computational Transportation Science, IWCTS@SIGSPATIAL 2016, Burlingame, California, USA, October 31 - November 3, 2016*. Ed. by Stephan Winter, Gautam S. Thakur, and Nicole Ronald. ACM, 2016, pp. 49–54. URL: https://doi.org/10.1145/3003965.3003966.

[160] Nabil Mohamed Drawil, Haitham Amar, and Otman Basir. “GPS Localization Accuracy Classification: A Context-Based Approach”. In: *IEEE Trans. Intelligent Transportation Systems* 14.1 (2013), pp. 262–273. URL: https://doi.org/10.1109/TITS.2012.2213815.

[161] Deboleena Sadhukhan and Madhuchhanda Mitra. “R-peak detection algorithm for ECG using double difference and RR interval processing”. In: *Procedia Technology* 4 (2012), pp. 873–877.

[162] Ryan S McGinnis et al. “Quantifying the effects of load carriage and fatigue under load on sacral kinematics during countermovement vertical jump with IMU-based method”. In: *Sports Engineering* 19.1 (2016), pp. 21–34. URL: https://doi.org/10.1007/s12283-015-0185-3%22.

[163] Emiliano De Cristofaro and Gene Tsudik. “Practical Private Set Intersection Protocols with Linear Complexity”. In: *Financial Cryptography and Data Security, 14th International Conference, FC 2010, Tenerife, Canary Islands, Spain, January 25-28, 2010, Revised Selected Papers*. 2010, pp. 143–159. URL: https://doi.org/10.1007/978-3-642-14577-3%5C_13.

[164] C. Carl Robusto. “The cosine-haversine formula”. In: *The American Mathematical Monthly* 64.1 (1957), pp. 38–40.

[165] Fan Bai, Narayanan Sadagopan, and Ahmed Helmy. “The IMPORTANT framework for analyzing the Impact of Mobility on Performance Of RouTing protocols for Adhoc NeTworks”. In: *Ad Hoc Networks* 1.4 (2003), pp. 383–403. URL: https://doi.org/10.1016/S1570-8705(03)00040-4.

[166] Reza Shokri et al. “Protecting location privacy: optimal strategy against localization attacks”. In: *the ACM Conference on Computer and Communications Security, CCS'12, Raleigh, NC, USA, October 16-18, 2012*. Ed. by Ting Yu, George Danezis, and Virgil D. Gligor. ACM, 2012, pp. 617–627. URL: https://doi.org/10.1145/2382196.2382261.

[167] Shaoqing Ren et al. “Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks”. In: *IEEE Trans. Pattern Anal. Mach. Intell.* 39.6 (2017), pp. 1137–1149. URL: https://doi.org/10.1109/TPAMI.2016.2577031.

[168] Christian Vaas, Panos Papadimitratos, and Ivan Martinovic. “Increasing Mix-Zone Efficacy for Pseudonym Change in VANETs using Chaff Messages”. In: *Proceedings of the 11th ACM Conference on Security & Privacy in Wireless and Mobile Networks, WiSec 2018, Stockholm, Sweden, June 18-20, 2018*. Ed. by Panos Papadimitratos, Kevin Butler, and Christina Pöpper. ACM, 2018, pp. 287–288. URL: http://doi.acm.org/10.1145/3212480.3226103.

[169] Christian Vaas et al. “Tracking Vehicles Through Encrypted Mix-Zones Using Physical Layer Properties”. In: *2018 IEEE Vehicular Networking Conference, VNC 2018, Taipei, Taiwan, December 5-7, 2018*. IEEE, 2018, pp. 1–2.

[170] Christian Vaas et al. “Nowhere to hide? Mix-Zones for Private Pseudonym Change using Chaff Vehicles”. In: *2018 IEEE Vehicular Networking Conference, VNC 2018, Taipei, Taiwan, December 5-7, 2018*. IEEE, 2018, pp. 1–9.

[171] Stéphanie Lefevre et al. “Impact of V2X privacy strategies on Intersection Collision Avoidance systems”. In: *2013 IEEE Vehicular Networking Conference, Boston, MA, USA, December 16-18, 2013*. IEEE, 2013, pp. 71–78. URL: https://doi.org/10.1109/VNC.2013.6737592.

[172] Xinxin Liu et al. “Traffic-aware multiple mix zone placement for protecting location privacy”. In: *Proceedings of the IEEE INFOCOM 2012, Orlando, FL, USA, March 25-30, 2012*. Ed. by Albert G. Greenberg and Kazem Sohraby. IEEE, 2012, pp. 972–980. URL: https://doi.org/10.1109/INFCOM.2012.6195848.

[173] Lara Codeca, Raphaël Frank, and Thomas Engel. “Luxembourg SUMO Traffic (LuST) Scenario: 24 hours of mobility for vehicular networking research”. In: *2015 IEEE Vehicular Networking Conference, VNC 2015, Kyoto, Japan, December 16-18, 2015*. IEEE, 2015, pp. 1–8. URL: https://doi.org/10.1109/VNC.2015.7385539.

[174] Balaji Palanisamy and Ling Liu. “Attack-Resilient Mix-zones over Road Networks: Architecture and Algorithms”. In: *IEEE Trans. Mob. Comput.* 14.3 (2015), pp. 495–508. URL: https://doi.org/10.1109/TMC.2014.2321747.

[175] John Krumm. “Inference Attacks on Location Tracks”. In: *Pervasive Computing, 5th International Conference, PERVASIVE 2007, Toronto, Canada, May 13-16, 2007, Proceedings*. Ed. by Anthony LaMarca, Marc Langheinrich, and Khai N. Truong. Vol. 4480. Lecture Notes in Computer Science. Springer, 2007, pp. 127–143. URL: https://doi.org/10.1007/978-3-540-72037-9%5C_8.

[176] Chen Chen et al. “TARANET: Traffic-Analysis Resistant Anonymity at the Network Layer”. In: *2018 IEEE European Symposium on Security and Privacy, EuroS&P 2018, London, United Kingdom, April 24-26, 2018*. IEEE, 2018, pp. 137–152. URL: https://doi.org/10.1109/EuroSP.2018.00018.

[177] Bin Fan et al. “Cuckoo Filter: Practically Better Than Bloom”. In: *Proceedings of the 10th ACM International on Conference on emerging Networking Experiments and Technologies, CoNEXT 2014, Sydney, Australia, December 2-5, 2014*. Ed. by Aruna Seneviratne et al. ACM, 2014, pp. 75–88. URL: http://doi.acm.org/10.1145/2674005.2674994.

[178] Hamid Noroozi, Mohammad Khodaei, and Panos Papadimitratos. “VPKIaaS: A Highly-Available and Dynamically-Scalable Vehicular Public-Key Infrastructure”. In: *Proceedings of the 11th ACM Conference on Security & Privacy in Wireless and Mobile Networks, WiSec 2018, Stockholm, Sweden, June 18-20, 2018*. Ed. by Panos Papadimitratos, Kevin Butler, and Christina Pöpper. ACM, 2018, pp. 302–304. URL: http://doi.acm.org/10.1145/3212480.3226100.

[179] Yipin Sun et al. “Mix-zones optimal deployment for protecting location privacy in VANET”. In: *Peer-to-Peer Networking and Applications* 8.6 (2015), pp. 1108–1121. URL: https://doi.org/10.1007/s12083-014-0269-z.

[180] Julien Freudiger, Reza Shokri, and Jean-Pierre Hubaux. “On the Optimal Placement of Mix Zones”. In: *Privacy Enhancing Technologies, 9th International Symposium, PETS 2009, Seattle, WA, USA, August 5-7, 2009. Proceedings*. Ed. by Ian Goldberg and Mikhail J. Atallah. Vol. 5672. Lecture Notes in Computer Science. Springer, 2009, pp. 216–234. URL: https://doi.org/10.1007/978-3-642-03168-7%5C_13.

[181] Michael Feiri et al. “PREparing SEcuRe VEhicle-to-X Communication Systems”. In: (Deliverable 3.2, FOT Trial 2 Results, 2015). URL: https://www.preserve-project.eu/deliverables.

[182] Boris Danev, Davide Zanetti, and Srdjan Capkun. “On physical-layer identification of wireless devices”. In: *ACM Comput. Surv.* 45.1 (2012), 6:1–6:29. URL: https://doi.org/10.1145/2379776.2379782.

[183] Anadi Chaman et al. “Ghostbuster: Detecting the Presence of Hidden Eavesdroppers”. In: *Proceedings of the 24th Annual International Conference on Mobile Computing and Networking, MobiCom 2018, New Delhi, India, October 29 - November 02, 2018*. Ed. by Rajeev Shorey et al. ACM, 2018, pp. 337–351. URL: http://doi.acm.org/10.1145/3241539.3241580.

[184] Marc Roeschlin et al. “Bionyms: Driver-centric Message Authentication using Biometric Measurements”. In: *2018 IEEE Vehicular Networking Conference, VNC 2018, Taipei, Taiwan, December 5-7, 2018*. IEEE, 2018, pp. 1–2.

[185] P. Remyakrishnan and C. Tripti. “A Novel Approach for Enhancing the Security of User Authentication in VANET Using Biometrics”. In: *Emerging ICT for Bridging the Future - Proceedings of the 49th Annual Convention of the Computer Society of India CSI Volume 2*. Berlin, Germany: Springer, 2015, pp. 299–306.

[186] Keun-Ho Lee and Soo Kyun Kim. “Authentication Scheme based on Biometric Key for VANET Information System in M2M Application Service”. In: *Appl. Math* 9 (2015), pp. 645–651.

[187] Lin Yao et al. “Biometrics-based Data Link Layer Anonymous Authentication in VANETs”. In: *Seventh International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing, IMIS 2013, Taichung, Taiwan, July 3-5, 2013*. Ed. by Leonard Barolli et al. IEEE Computer Society, 2013, pp. 182–187. URL: https://doi.org/10.1109/IMIS.2013.38.

[188] Arun Hegde. *Carsharing Market to witness a massive 34%+ growth over 2016-2024*. 2017.

[189] Warwick Goodall et al. “The rise of mobility as a service”. In: *Deloitte Rev* 20 (2017), pp. 112–129.

[190] Ivan Damgård and Eiichiro Fujisaki. “A Statistically-Hiding Integer Commitment Scheme Based on Groups with Hidden Order”. In: *Advances in Cryptology - ASIACRYPT 2002, 8th International Conference on the Theory and Application of Cryptology and Information Security, Queenstown, New Zealand, December 1-5, 2002, Proceedings*. Ed. by Yuliang Zheng. Vol. 2501. Lecture Notes in Computer Science. Springer, 2002, pp. 125–142. URL: https://doi.org/10.1007/3-540-36178-2%5C_8.

[191] Claus-Peter Schnorr. “Efficient Identification and Signatures for Smart Cards”. In: *Advances in Cryptology - CRYPTO ’89, 9th Annual International Cryptology Conference, Santa Barbara, California, USA, August 20-24, 1989, Proceedings*. Ed. by Gilles Brassard. Vol. 435. Lecture Notes in Computer Science. Springer, 1989, pp. 239–252. URL: https://doi.org/10.1007/0-387-34805-0%5C_22.

[192] Christian Rathgeb and Andreas Uhl. “A survey on biometric cryptosystems and cancelable biometrics”. In: *EURASIP J. Information Security* 2011 (2011), p. 3. URL: https://doi.org/10.1186/1687-417X-2011-3.

[193] Ivan Martinovic et al. “Pulse-Response: Exploring Human Body Impedance for Biometric Recognition”. In: *ACM Trans. Priv. Secur.* 20.2 (2017), 6:1–6:31. URL: https://doi.org/10.1145/3064645.

[194] Marc Roeschlin et al. “Generating Secret Keys from Biometric Body Impedance Measurements”. In: *Proceedings of the 2016 ACM on Workshop on Privacy in the Electronic Society, WPES@CCS 2016, Vienna, Austria, October 24 - 28, 2016*. Ed. by Edgar R. Weippl, Stefan Katzenbeisser, and Sabrina De Capitani di Vimercati. ACM, 2016, pp. 59–69. URL: http://dl.acm.org/citation.cfm?id=2994626.

[195] OMRON Healthcare. *Weight Management - Frequently asked questions*. 2017.

[196] Daniel Krajzewicz et al. “Recent development and applications of SUMO-Simulation of Urban MObility”. In: *International Journal On Advances in Systems and Measurements* 5 (2012), pp. 128–138.

[197] Stephan Eichler. “Strategies for pseudonym changes in vehicular ad hoc networks depending on node mobility”. In: *Intelligent Vehicles Symposium, 2007 IEEE*. IEEE. 2007, pp. 541–546.

[198] David Shepardson. *Tesla and other automakers are desperately trying to make sure drivers keep their hands on the wheel*. 2017.

[199] Simon Eberz et al. “Preventing Lunchtime Attacks: Fighting Insider Threats With Eye Movement Biometrics”. In: *22nd Annual Network and Distributed System Security Symposium, NDSS 2015, San Diego, California, USA, February 8-11, 2015*. The Internet Society, 2015. URL: https://www.ndss-symposium.org/ndss2015/preventing-lunchtime-attacks-fighting-insider-threats-eye-movement-biometrics.

[200] Andreas Riener and Alois Ferscha. “Supporting implicit human-to-vehicle interaction: Driver identification from sitting postures”. In: *The first annual international symposium on vehicular computing systems (isvcs 2008)*. Vol. 10. USA: isvcs, 2008, p. 10.

[201] Tyler Kaczmarek, Ercan Ozturk, and Gene Tsudik. “Assentication: User De-authentication and Lunchtime Attack Mitigation with Seated Posture Biometric”. In: *Applied Cryptography and Network Security - 16th International Conference, ACNS 2018, Leuven, Belgium, July 2-4, 2018, Proceedings*. Ed. by Bart Preneel and Frederik Vercauteren. Vol. 10892. Lecture Notes in Computer Science. Springer, 2018, pp. 616–633. URL: https://doi.org/10.1007/978-3-319-93387-0%5C_32.

[202] Fahim Sufi, Ibrahim Khalil, and Jiankun Hu. “ECG-Based Authentication”. In: *Handbook of Information and Communication Security*. Ed. by Peter P. Stavroulakis and Mark Stamp. Springer, 2010, pp. 309–331. URL: https://doi.org/10.1007/978-3-642-04117-4%5C_17.

[203] John D. Lee and Katrina A. See. “Trust in Automation: Designing for Appropriate Reliance”. In: *Human Factors* 46.1 (2004), pp. 50–80. URL: https://doi.org/10.1518/hfes.46.1.50.30392.