
# PREparing SEcuRe VEhicle-to-X Communication Systems

## Deliverable 5.3

## Deployment Issues Report V3

| | |
|---|---|
| **Project:** | **PRESERVE** |
| **Project Number:** | **IST-269994** |
| **Deliverable:** | **D5.3** |
| **Title:** | **Deployment Issues Report V3** |
| **Version:** | **1.0** |
| **Confidentiality:** | **Public** |
| **Editor:** | **M. Laganà** |
| **Cont. Authors:** | **M. Laganà, J. Petit, C. Jouvray, M. Feiri, N. Bißmeyer** |
| **Date:** | **2013-12-20** |


**Part of the Seventh Framework Program**
**Funded by the EC-DG INFSO**

# Document History

| Version | Date | Main author | Summary of changes |
|---|---|---|---|
| v0.1 | 2013-11-18 | M. Laganà (KTH) | Initial version |
| v0.2 | 2013-11-26 | J. Petit (UT) | Privacy section |
| v0.3 | 2013-11-26 | C. Jouvray (Trialog) | Business models |
| v0.4 | 2013-11-28 | M. Feiri (UT) | Security Architecture |
| v0.5 | 2013-11-29 | M. Laganà (KTH) | Identity Management |
| v0.6 | 2013-12-03 | N. Bißmeyer (Fraunhofer SIT) | Misbehavior Detection and Prevention |
| v1.0 | 2013-12-20 | M. Laganà (KTH), J. Petit (UT) | Final version |

| **Approval** | | |
|---|---|---|
| | Name | Date |
| Prepared | Marcello Laganà, Jonathan Petit | 2013-12-20 |
| Reviewed | All Project Partners | 2014-01-28 |
| Authorized | Frank Kargl | 2014-02-04 |

| **Circulation** | |
|---|---|
| Recipient | Date of submission |
| Project Partners | 2014-02-04 |
| European Commission | 2014-02-04 |

# Contents

# List of Figures

# List of Tables

# 1 Glossary

| Abbrev | Synonyms | Description | Details |
|---|---|---|---|
| **API** | | Application Programming Interface | An API is a particular set of specifications that software programs can follow to communicate with each other. |
| **AU** | | Application Unit | Hardware unit in an ITS station running the ITS applications |
| **ASN.1** | | Abstract Syntax Notation One | ASN.1 is a standard and flexible notation that describes data structures for representing, encoding, transmitting, and decoding data. |
| **CA** | | Certificate Authority | A CA is an entity that issues digital certificates. |
| **CAM** | | Cooperative Awareness Message | CAMs are sent by vehicles multiple times a second (typically up to 10 Hz), they are broadcasted unencrypted over a single-hop and thus receivable by any receiver within range. They contain the vehicle's current position and speed, along with information such as steering wheel orientation, brake state, and vehicle length and width. |
| **CAN** | | Controller Area Network | A CAN is a vehicle bus standard designed to allow microcontrollers and on-board devices to communicate with each other. |
| **CCM** | | Communication Control Module | Module responsible for protecting on-board communication. Originates from the EVITA project. |
| **CCU** | | Communication & Control Unit | Hardware unit in an ITS station running the communication stack |
| **CE** | | Consumer Electronics | Electronic devices like smartphone or MP3 player of the vehicle driver or a passenger |

| Abbrev | Synonyms | Description | Details |
|---|---|---|---|
| **CL** | | Convergence Layer | Module that connects the external on-board entities (e.g. communication stack or applications) to the PRESERVE Vehicle Security Subsystem (VSS) |
| **CPU** | | Central Processing Unit | |
| **CRC** | | Cyclic Redundancy Code | Is used to produce a checksum in order to detect errors in data storage or transmission. |
| **CRS** | | Cryptographic Services | Module acting as proxy for accessing different cryptographic algorithm implementations. Originates from the EVITA project |
| **DoS** | | Denial of Service | A DoS is a form of attack on a computer system or networks. |
| **DENM** | DNM | Decentralized Environmental Notification Message | A DENM transmission is triggered by a cooperative road hazard warning application, providing information to other ITS stations about a specific driving environment event or traffic event. The ITS station that receives the DENM is able to provide appropriate HMI information to the end user, who makes use of these information or takes actions in its driving and traveling. Fehler: Referenz nicht gefunden |
| **EAM** | | Entity Authentication Module | Module responsible for ensuring entity authentication of in-vehicle components. Originates from the EVITA project |
| **ECC** | | Elliptic Curve Cryptography | ECC is an approach to public-key cryptography based on the algebraic structure of elliptic curves over finite fields. |
| **ECU** | | Electronic Control Unit | |
| **ECR** | | ECU configuration register | Register used for secure boot and authenticated boot inside the HSM (similar to platform configuration register inside a TPM) |
| **FOT** | | Field Operational Test | |

| Abbrev | Synonyms | Description | Details |
|---|---|---|---|
| **G5A** | | ITS road safety communication (802.11p) | Frequency band between 5.875 GHz and 5.905 GHz - reserved for ITS road safety communication |
| **G5B** | | ITS non-safety communication (802.11p) | Frequency band between 5.855 GHz and 5.875 GHz - reserved for ITS road non-safety communication |
| **G5C** | C-WLAN | 5GHz WLAN communication (802.11a) | |
| **GNSS** | GPS | Global Navigation Satellite System | Generic term for an Global navigation satellite system (GPS, GLONAS, Galileo) |
| **HMI** | | Human-Machine Interface | |
| **HSM** | | Hardware Security Module | |
| **HU** | | Head-Unit | |
| **I2V** | I2C | Infrastructure-to-Vehicle | Communication between infrastructure components like roadside units and vehicles |
| **I2I** | | Infrastructure-to-Infrastructure | Communication between multiple infrastructure components like roadside units |
| **ICS** | | ITS Central Station | ITS station in a central ITS subsystem |
| **ILP** | | Inter Layer Proxy | Component introduced by the SeVeCom project, that captures and allows modification of messages between different layers of a communication stack |
| **IDK** | Module Authentication Key | Device Identity Key | The Device Identity Key is introduced by EVITA and is used for HSM identification. The IDK can also be certified by a manufacturer authentication key. |
| **IMT** | GSM, GPRS, UMTS | Public cellular services (2G, 3G, ...) | |
| **IPR** | | Intellectual Property Right | |

| Abbrev | Synonyms | Description | Details |
|---|---|---|---|
| **ITS** | | Intelligent Transportation Systems | Intelligent Transport Systems (ITS) are systems to support transportation of goods and humans with information and communication technologies in order to efficiently and safely use the transport infrastructure and transport means (cars, trains, planes, ships). |
| **ITS-S** | | ITS Station | Generic term for any ITS station like vehicle station, roadside unit, ... |
| **IDM** | | ID & Trust Management Module | Module responsible for ID management originating from SeVeCom project. |
| **IVC** | ITSC, ITS Communications | Inter-Vehicle Communication | Combination of V2V and V2I |
| **IVS** | OBU | ITS Vehicle Station | The term "vehicle" can also be used within PRESERVE |
| **LDM** | Environment Table | Local Dynamic Map | Local geo-referenced database containing a V2X-relevant image of the real world |
| **LTC** | | Long-Term Certificate | PRESERVE realization of an ETSI Enrolment Credential. The long-term certificate authenticates a stations within the PKI, e.g., for PC refill and may contain identification data and properties. |
| **LTCA** | | Long-Term Certificate Authority | PRESERVE realization of an ETSI Enrollment Credential Authority that is part of the PKI and responsible for issuing long-term certificates. |
| **MAC** | | Media Access Control | The MAC data communication protocol sub-layer is a sublayer of the Data Link Layer specified in the seven-layer OSI model. |
| **OBD** | | On-Board Diagnosis | OBD is a generic term referring to a vehicle's self-diagnostic and reporting capability that can be used by a repair technician to access the vehicles sub-systems. |

| Abbrev | Synonyms | Description | Details |
|---|---|---|---|
| **OEM** | | Original Equipment Manufacturer | Refers to an generic car manufacturer |
| **OBU** | IVS | On-Board Unit | An OBU is part of the V2X communication system at an ITS station. In different implementations different devices are used (e.g. CCU and AU) |
| **PAP** | | Policy Administration Point | Module related to the PDM originating from EVITA project |
| **PC** | Short Term Certificate | Pseudonym Certificate | A short term certificate authenticates stations in G5A communication and contains data reduced to a minimum. |
| **PCA** | | Pseudonym Certificate Authority | Certificate authority entity in the PKI that issues pseudonym certificates |
| **PDM** | | Policy Decision Module | Module responsible for enforcing the use of policies originating from EVITA project |
| **PDP** | | Policy Decision Point | Module related to the Policy Decision Module originating from EVITA project |
| **PeRA** | | Privacy-enforcing Runtime Architecture | Module responsible for enforcing privacy protection policies originating from PRECIOSA project |
| **PEP** | | Policy Enforcement Point | Module related to the Policy Decision Module originating from EVITA project |
| **PIM** | | Platform Integrity Module | Module responsible for ensuring in-vehicle component integrity originating from EVITA project |
| **PKI** | | Public Key Infrastructure | A PKI is a set of hardware, software, policies, and procedures needed to create, manage, distribute, use, store, and revoke digital certificates. |
| **PMM** | | Pseudonym Management Module | Module responsible for management of the station's pseudonym certificates originating from SeVeCom project |
| **RSU** | IRS, ITS Roadside Station | Roadside Unit | A RSU is a stationary or mobile ITS station at the roadside acting as access point to the infrastructure. |

| Abbrev | Synonyms | Description | Details |
|---|---|---|---|
| **SAP** | | Service Access Point | Informative functional specification that enables the interconnection of different component implementations. |
| **SM** | | Security Manager | Module responsible for securing the V2X communication with external ITS stations originating from SeVeCom project |
| **SCM** | | Secure Communication Module | A generic name for the complete secure communication stack |
| **SEP** | | Security Event Processor | Module responsible for security event management (e.g. checking message plausibility, station reputation calculation) |
| **TPM** | | Trusted Platform Module | A TPM is both, the name of a published specification detailing a secure crypto-processor that can store cryptographic keys, as well as the general name of implementations of that specification, often called the "TPM chip" or "TPM Security Device". |
| **UML** | | Unified Modeling Language | UML is an object modeling and specification language used in software engineering. |
| **UTC** | | Coordinated Universal Time | UTC is the primary time standard by which the world regulates clocks and time. |
| **V2I** | C2I | Vehicle-to-Infrastructure | Direct vehicle to roadside infrastructure communication using a wireless local area network |
| **V2V** | C2C | Vehicle-to-Vehicle | Direct vehicle(s) to vehicle(s) communication using a wireless local area network |
| **V2X** | C2X | Vehicle-to-Vehicle (V2V) and/or Vehicle-to-Infrastructure (V2I) | Direct vehicle(s) to vehicle(s) or vehicle(s) to infrastructure communication using a wireless local area network |
| **VIN** | | Vehicle Identification Number | Unique serial number of a vehicle |
| **VSA** | | Vehicle Security Architecture | General outcome of PRESERVE work package 1 |

| Abbrev | Synonyms | Description | Details |
|---|---|---|---|
| **VSS** | | V2X Security Subsystem | Close-to-market implementation of the PRESERVE VSA that is the outcome of PRESERVE work package 2 |
| **WLAN** | | Wireless Local Area Network | |
| **XML** | | Extensible Markup Language | XML is a set of rules for encoding documents in machine-readable form. |

# 2 Introduction

The Work Package 5 investigates the major security and privacy related aspects in ITS that have not been taken into account, and thus, have not been sufficiently addressed. These aspects also include issues related to the market introduction of V2X security systems.

The focus in this report is on forward looking issues, beyond the PRESERVE architecture and security subsystem. Therefore, in this deliverable, we investigate the deployment issues of PRESERVE. More specifically, we investigate how the PRESERVE platform is seen and received by the community, and discuss valid business model for the project (see Section 3). We provide preliminary results for the PRESERVE questionnaire that was created and disseminated to stakeholders in automotive industry and beyond to analyze their awareness of security and privacy. Moreover, we continue providing directions regarding business models of PRESERVE results, that could be exploited by the partners.

One of the PRESERVE objectives is to create an integrated V2X Security Architecture (VSA). Although non-fundamental, misbehavior detection is a key aspect of such architecture. In Section 4 we introduce the state-of-the-art regarding misbehavior detection and present a logic framework. Moreover, we conduct an experimental analysis on how to detect and prevent an internal attacker to cause harm.

Although privacy-preserving mechanisms are one of the requirements for the vehicular communications, their impact for example on the Intersection Collision Avoidance system needs to be evaluated. Furthermore, the vehicular universe is expanding with the addition of new vehicle types, such as the Electric Vehicles (EVs), and new privacy challenges arise. Therefore, new privacy-preserving protocols shall be designed. Those privacy-related discussions are presented in Section 5.

User's privacy must be preserved while misbehaving entities should be detected, and eventually evicted from the system. Therefore, in Section 6, we present identity management systems that allow privacy-preserving capabilities as long as accountability features.

Finally, in Section 7 we extend the security architecture with new features, and we investigate the impact of the security on cooperative awareness.

The results presented herein correspond to the tasks 5200 and 5110 of the PRESERVE Description of Work. In addition, the WP5 reports provide a track record of all related research output. Accordingly, the V2X Security Subsystem (VSS) does not integrate all schemes presented in this deliverable: the details regarding VSS, notably its first version, are available in deliverables of WP2 and WP4, the field trial related material in deliverables of WP3, and the individual exploitation of the business model will be presented in the

deliverables of WP6. It is expected that the second version of the VSS will integrate some schemes and elements that are results of the ongoing WP5 work.

# 3 Adoption of PRESERVE

## 3.1 Broadening awareness on the PRESERVE platform

There is a consensus being formed, in terms of basic technological aspects for security and privacy in ITS. Nonetheless, many questions concerning the actual deployment of these systems are not addressed yet. In addition, issues such as product life-cycles and costs for ITS products and services have to be defined, so that vehicular communication solutions can be brought to market. These are in fact important factors for the PRESERVE project and more generally for the ITS community.

In order to gauge the perception of the broader ITS community regarding the security and privacy needs for ITS and the PRESERVE architecture, we have designed and disseminated a questionnaire that seeks answers to the above and serves as an extension of the investigations related to this work package of PRESERVE. This section provides an overview of the structure of our survey along with the methodology for its design. The questionnaire can be found on the PRESERVE website[1].

We are currently in the process of collecting responses. Rather than including here limited results and thus providing a limited analysis, we shall update this report in the Deliverable D5.3.

### 3.1.1 Overview of the Survey

Our survey is designed in a way that no prior knowledge of the responders is presumed. We begin by asking the responder to provide us with input on her background. This helps us to better analyse the responses and weight them accordingly. Moreover, we treat each individual response as anonymous and strictly confidential. We emphasize that answers reflect the opinions of the individual responder alone and not of the institutions they represent. Similarly, once the responses are collected the resulting analysis will present aggregate responses. We use three types of questions; *multiple choice*, *free text* and *matrix questions*. For the latter ones, we utilize a scale from $0$ (low) to $4$ (high). The questionnaire comprises six sections:

- **Introductory Questions**: This section contains general questions concerning the background of the responder in terms of security and privacy for ITS. In addition, we try to capture the understanding of the responder on the PRESERVE architecture.

[1] http://www.preserve-project.eu/node/43

- **Questions on Safety Applications**: These questions focus on security and privacy requirements for specific safety applications as defined in the survey. We also inquire on the suitability of the PRESERVE architecture for protecting these applications.
- **Questions on Infotainment and Miscellaneous Applications**: These two sections focus on infotainment and miscellaneous applications. Similarly to the previous section, we are interested in the security and privacy requirements of these applications and in the applicability of PRESERVE's VSA for these application types.
- **Questions Regarding Financial Aspects**: These questions target responders whose role in the institutions they represent is of managerial nature.
- **Questions Regarding Technical Aspects**: This category contains questions of technical nature that target the part of the audience/responders with technical security and privacy expertise.

### 3.1.2 Survey Dissemination

We have created an on-line version of our survey which allows enhanced dissemination and analysis capabilities. The link to the survey has been uploaded on the web-page of the PRESERVE project [2] The survey disseminated to various responders such as standardization bodies and experts in the are of ITS. We gathered responding volunteers during the ITS World Congress held in Vienna from $22$ to $26$ of October $2012$. In addition, we advertised our survey during the proceedings of C2C-CC Forum held in Göteborg (Sweden) on $13$ and $14$ of November $2012$ and in the EIT-ICT Safe Mobility chapter [3]. We have continued with collaborating FOT projects, with a US-EU Harmonization Working Group, and select researchers in the broader ITS area.

### 3.1.3 Preliminary Analysis of the Results

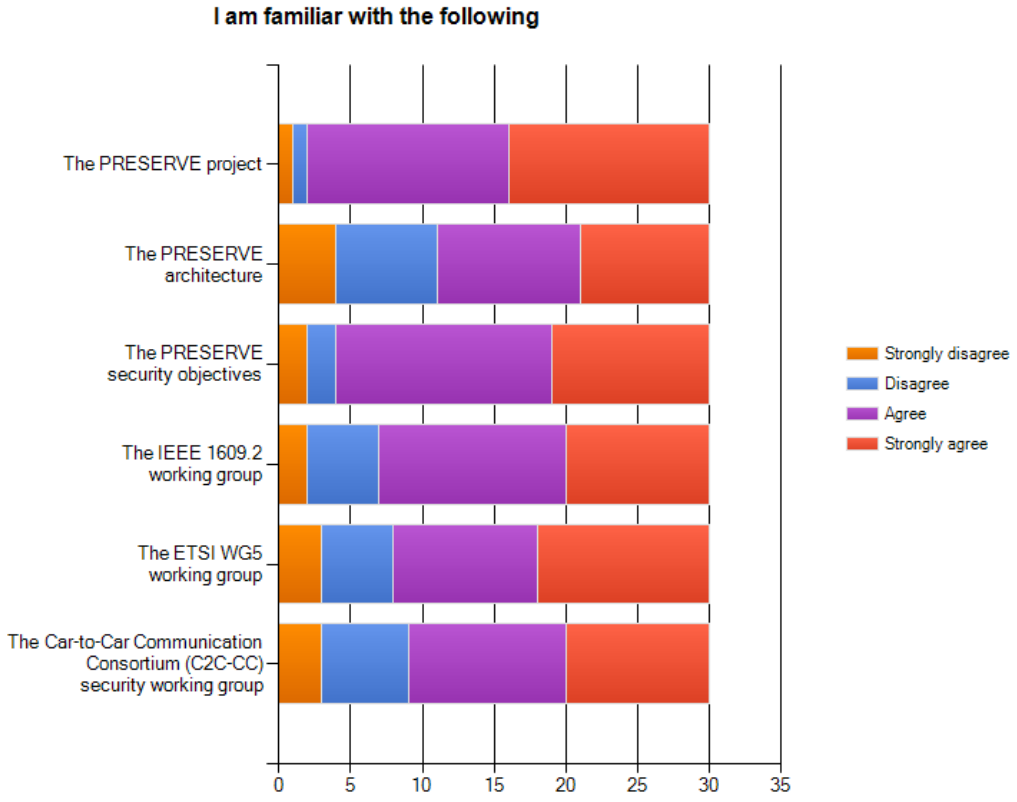
Since the process of collecting answers is still in progress, in this section we present a preliminary analysis of answers of the responders. We focus on the most representative questions of each section. A full analysis will be included in Deliverable 5.4.

#### 3.1.3.1 Introductory Questions

This section includes seven $(7)$ questions. The first questions (*Q1* and *Q2*) ask for the responders' personal information. Given we treat the answers of individuals anonymously, fields such as the responders' name, email and phone are only *optional*. The only pieces of information we require are the organization position and the organization type for the responder. Based on this question we can have an understanding of her background.

[2] http://preserve-project.eu/
[3] http://www.eitictlabs.eu/action-lines/intelligent-mobility-and-transportation-systems/

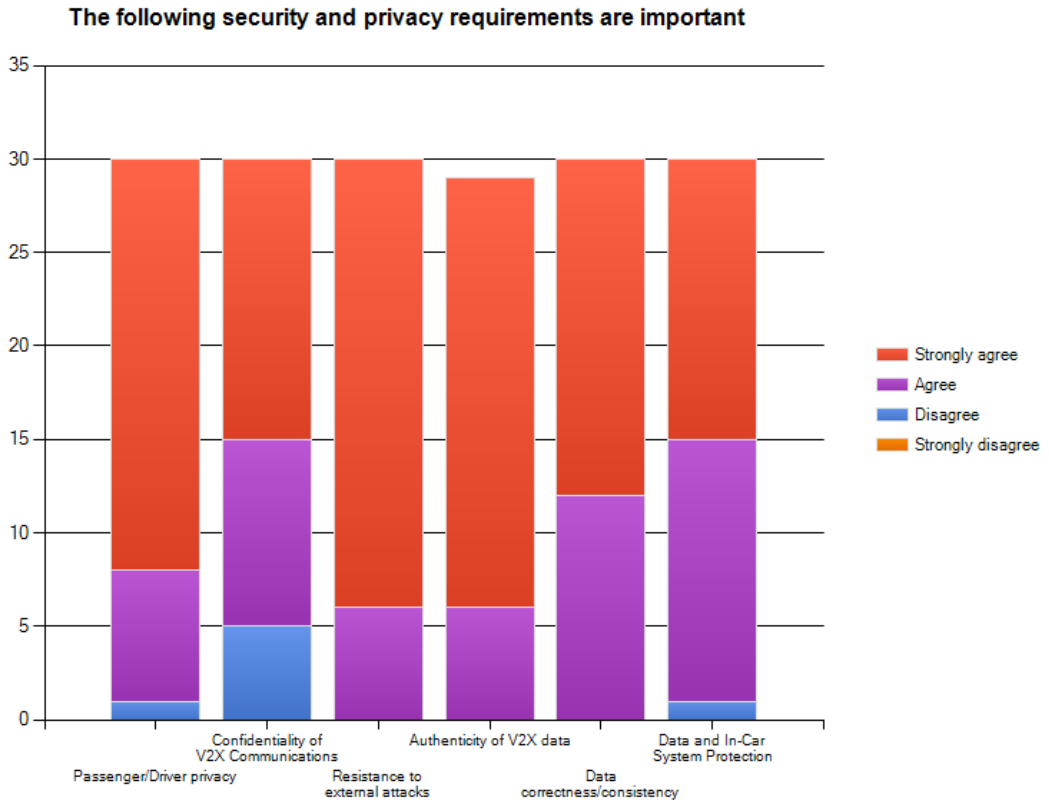
**Question 3** In *Q3*, we ask the responders about their familiarity with the PRESERVE project and various standardization bodies active in the area of ITS (IEEE 1609.2-WG[4], ETSI-WG5[5] and C2C-CC [2]). If the respondent is familiar with the above, she is considered to be a specialist when it comes to technical aspects for ITS and the answers will be analyzed accordingly. The following figure illustrates the received responses.




We can observe the majority of the responders are familiar with the PRESERVE project ($92\%$), the PRESERVE architecture ($63\%$) and the project's security objectives ($86\%$). Furthermore, the responders are familiar the aforementioned standardization bodies and working groups; $76\%$ of the responders knows the IEEE working group, $73\%$. For the ETSI-WG5 and the C2C-CC, the percentages are $73\%$ and $69\%$ respectively.

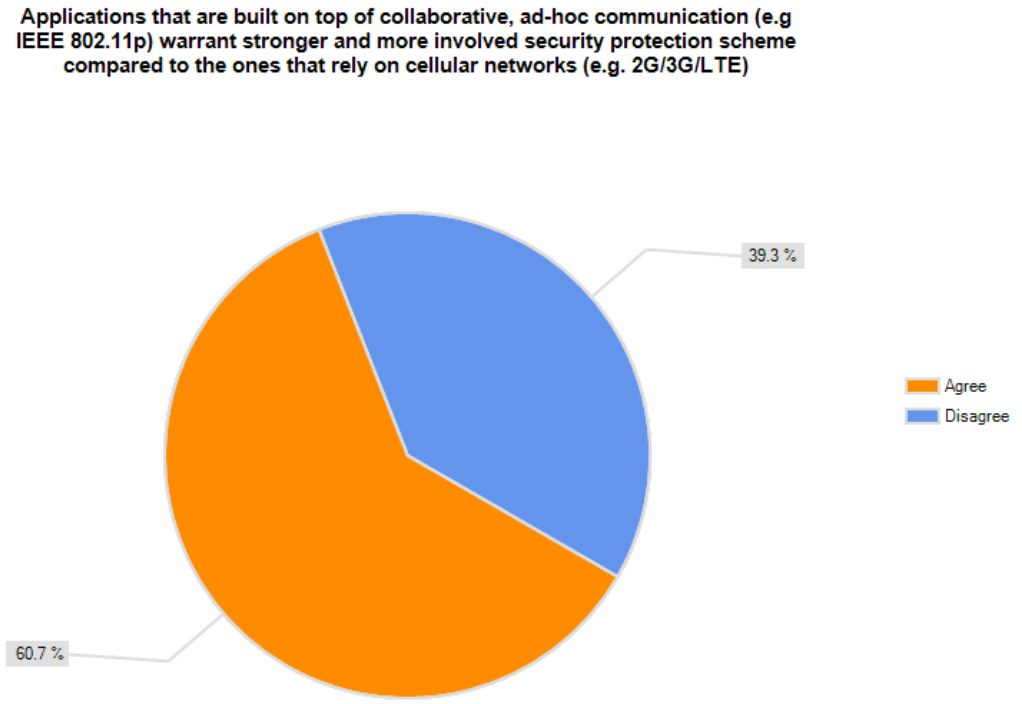
**Question 4** *Q4* asks the responders how important they consider security and privacy requirements to be. These requirements are extracted from the state-of-the-art research and the relevant technical literature.

[4] http://vii.path.berkeley.edu/1609_wave/
[5] http://www.etsi.org/website/technologies/intelligenttransportsystems.aspx




As the figure shows, the majority of responders considers passenger and privacy to be of paramount importance ($96\%$). This high percentage reflects the strong research interest for privacy preserving vehicular communications. The same holds with the rest of the requirements. More specifically, $83\%$ of the responders believes that ensuring the authenticity of V2X communications is a critical requirement. All responders agree that resilience against external attacks is an important requirement. The same consensus holds in the case of communication authenticity ($100\%$) and in-Car protection ($97\%$).

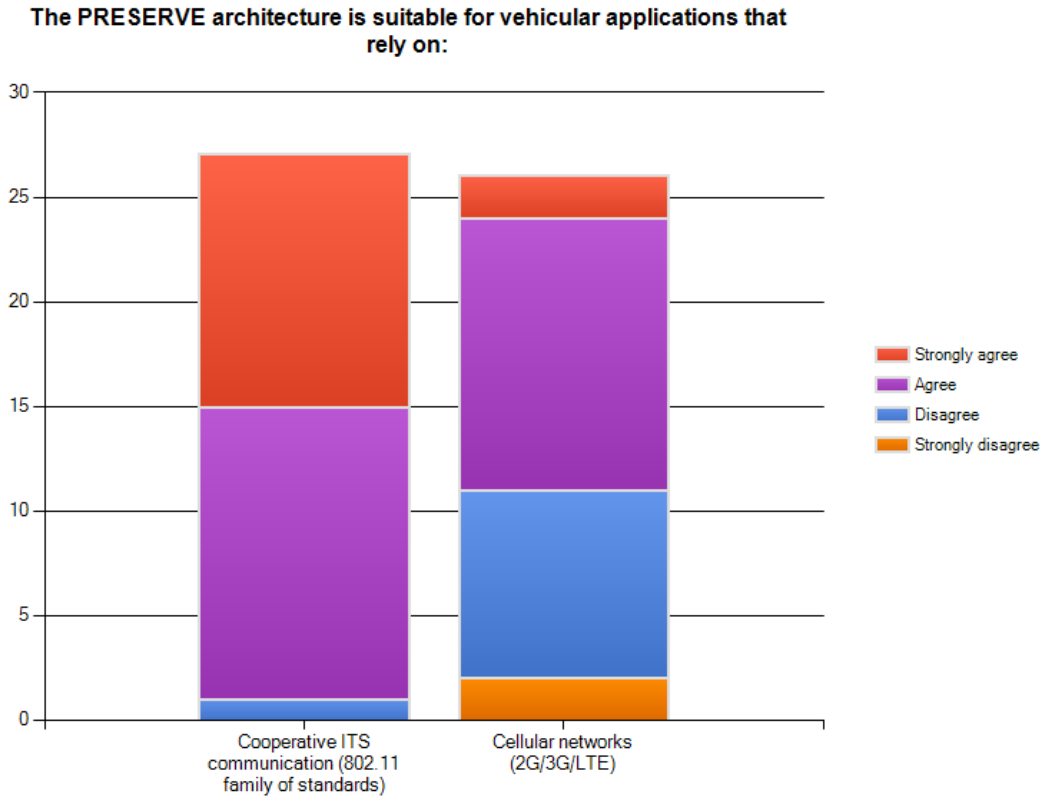
**Question 5** *Q5* tries to identify whether the broader ITS community considers applications built on top of collaborative, ad hoc communication (IEEE 802.11p) warrant stronger and more involved security protection scheme compared to the ones that rely on cellular networks (e.g., 2G/3G/LTE).



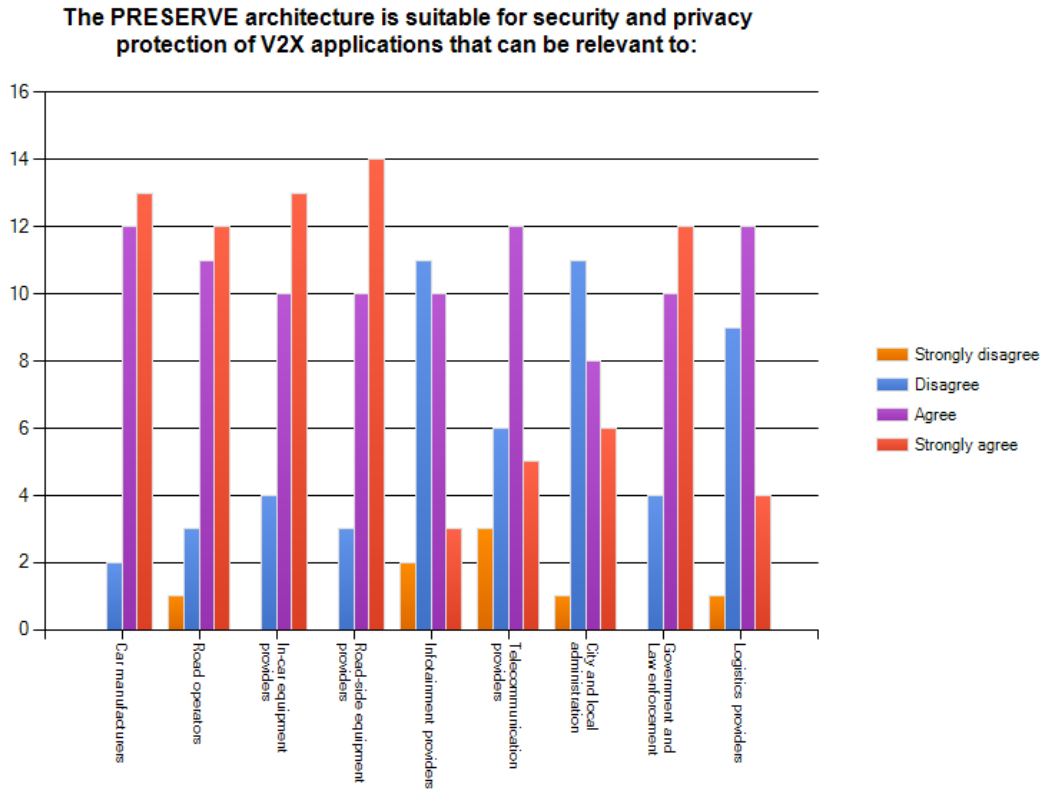

As the diagram shows, the majority of the responders considers that applications built on top of ad hoc communication schemes require stronger security protection compared to applications that rely on cellular networks. This result is an accordance with the results of the panel discussion that took place during the IEEE VNC 2011[6]

**Question 6** *Q5*, in *Q6* we ask the responders if they consider the results of PRESERVE applicable for ITS applications that built on cellular networks. As it can be seen in the following figure, $95\%$ of the responders agree that PRESERVE's architecture is applicable for applications built on top of 802.11p. Although this percentage decreases in the case of applications built on top of cellular networks (e.g., 3G, LTE), still the majority of the responders agrees that PRESERVE can ensure the security and privacy of such applications.

[6] http://www.ieee-vnc.org/2011/talks/panel.pdf




**Question 7** *Q7* asks the opinion of the responders regarding the applicability of PRESERVE to applications specific to various different domains.




The responders of this question agree that PRESERVE can meet the security and privacy requirements of a wide gamut of applications. The only exception is for applications relevant to *Telecommunication Providers* $(34\%)$ and providers of infotainment services $(50\%)$.

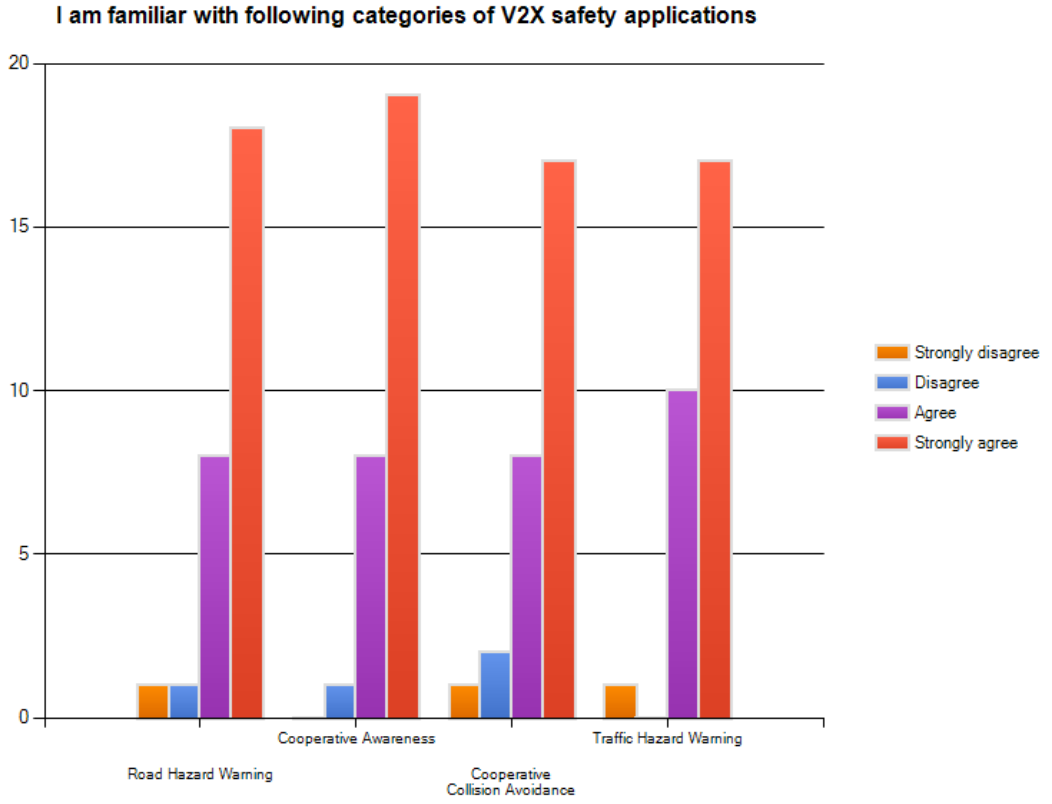
### 3.1.3.2 Safety Applications Questions

In this section, the survey focuses on safety applications. We consider the following list of safety applications:

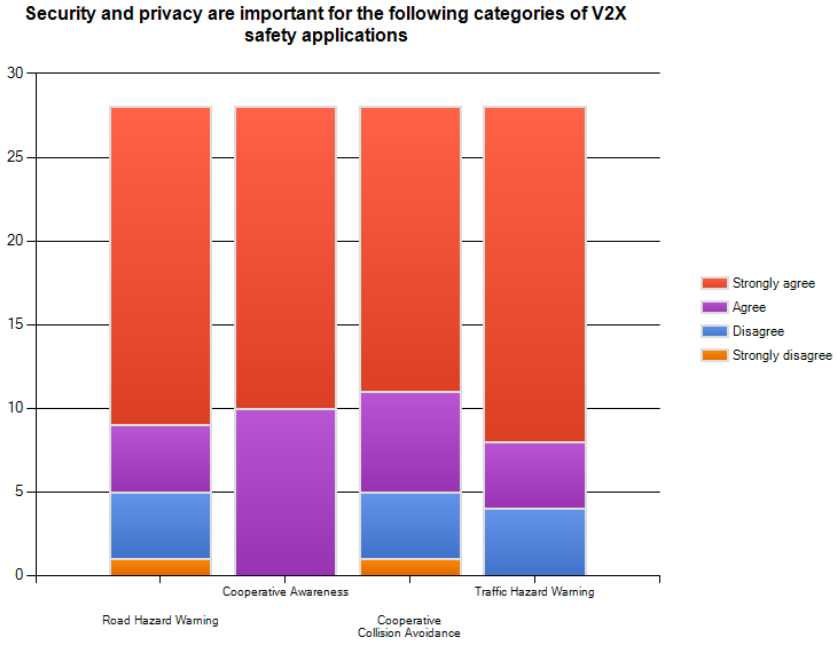
- **Road Hazard Warning**: Sudden slow-down warning, vehicle safety function out of normal condition warning
- **Cooperative Awareness**: Emergency vehicles notification, slow vehicle notification, motorcycle notification
- **Cooperative Collision Avoidance**: Vulnerable user warning
- **Traffic Hazard Warning**: Wrong way driving notification, stationary vehicle notification, traffic jam notification, signal violation notification

This section contains four $(4)$ questions whose purpose is to help us understand *if* and *how* PRESERVE's VSS can be utilized to guarantee the security and privacy requirements of the four safety applications presented above.

**Questions 8, 9** *Q8* and *Q9* probe the familiarity of the respondents concerning the security and privacy requirements of safety applications. As the core focus of PRESERVE is on safety applications, it is critical to understand the opinion of the ITS community concerning the suitability of PRESERVE for these applications. The following figure illustrates the answers of the responders to *Q8*.




The majority of the responders is familiar with the different types of safety applications. Furthermore, the responders agree that security and privacy are of paramount importance for safety applications, as the following figure shows.



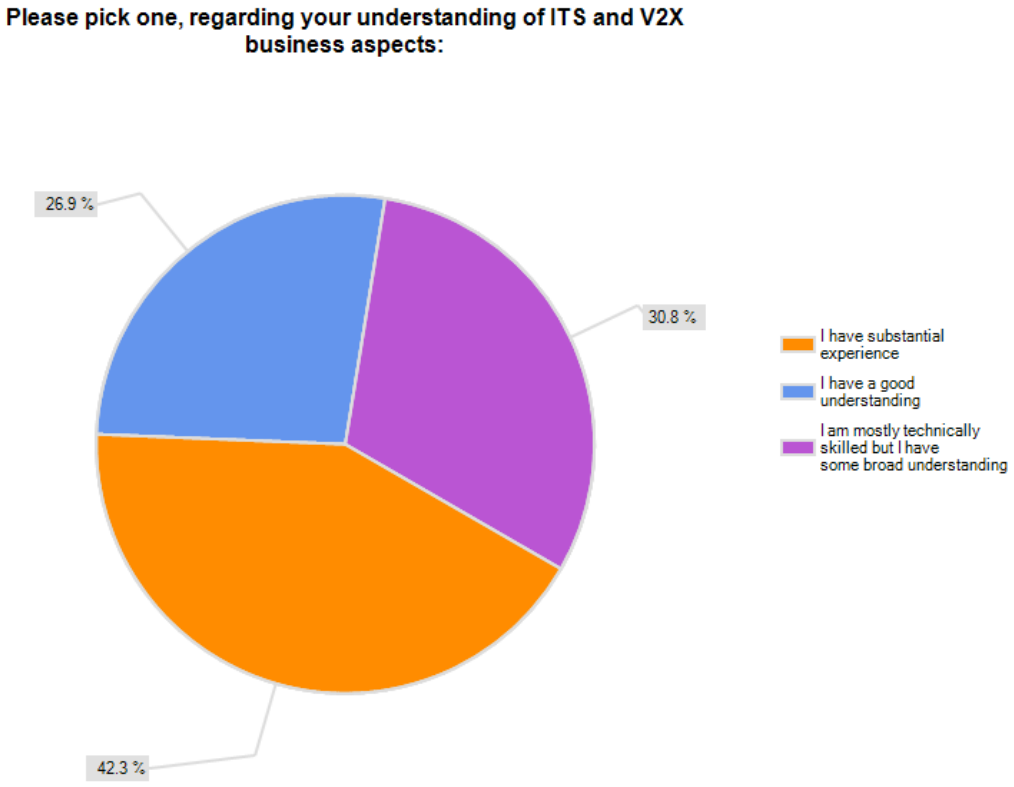

#### 3.1.3.3 Traffic Efficiency and Infotainment Applications Questions

These two sections follow the same structure and mentality with the previous one. Their difference is that they focus on traffic efficiency applications (Sec. 3 of the questionnaire) and infotainment applications (Sec. 4). To facilitate the answering of the questions in these sections we provide the responders with lists of traffic efficiency and infotainment applications according to [3]. The analysis of these sections will be included in Deliverable 5.4.

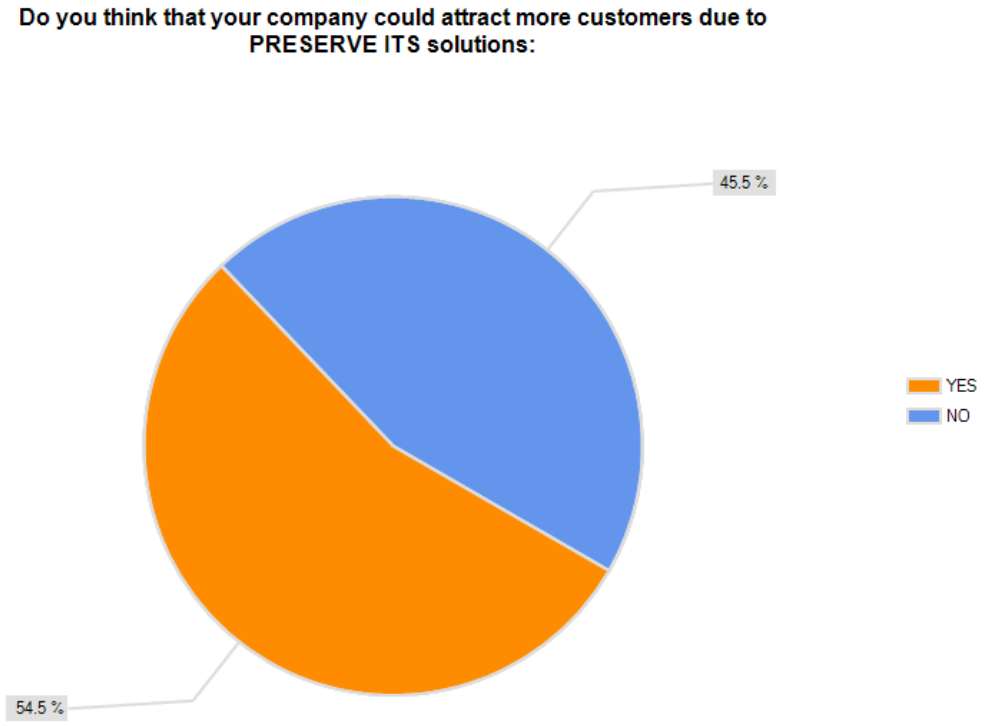
#### 3.1.3.4 Financial Aspects Questions

This section of the survey targets responders whose role in the company or the institution they represent is of managerial/business (non-technical) nature.

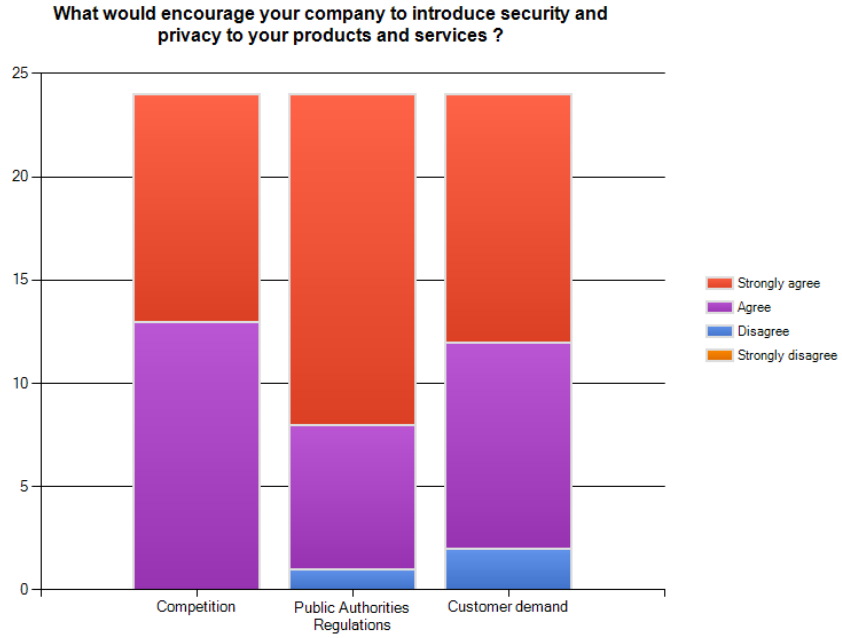
**Question 21** *Q21*, gauges the understanding of the respondents on the business aspects of ITS systems. As the following figure shows, the majority of the responders (69%) has either a substantial experience or a good understanding of the business aspects of ITS.




**Question 22** *Q22* asks the respondent's opinion on the potential commercial value of the PRESERVE ITS solution. The majority of responders ($54.5\%$) agrees on the potential commercial value of PRESERVE in the context of ITS.




**Question 23** *Q23* asks for the motives that drive organizations and institutions to introduce security and privacy solutions into their ITS related products and services.




As it can be seen from the figure, all of the responders believe that their organization will incorporate security and privacy in their products mostly due to the competition. In addition, $95\%$ of the responders answer that they also take into consideration the regulations originating from public authorities.

#### 3.1.3.5 Technical Aspects Questions

The final section of our survey is concerned with technical aspects of security and privacy for ITS. We begin by asking the responders in *Q32* about their technical background and understanding of technical aspects of security and privacy of ITS.

In *Q33* we require the responders to provide their input on the impact of a set of security and privacy threats. In questions *Q34*, *Q35* and *Q36* we ask the responder on the suitability of different cryptographic schemes in the context of safety, traffic efficiency and infotainment applications. We conclude this section with *Q37* which asks the responders to provide their input on the technical challenges towards the deployment of secure and privacy protecting ITS. An extensive analysis of this section will be included in Deliverable 5.4.

#### 3.1.3.6 Conclusions

From the initial answers, we can deduce that the community definitely values the security and privacy aspects of vehicular communications and considers PRESERVE as a valid proposal that meets these requirements. Nonetheless it is yet not clear whether or not the

presence of PRESERVE will create added value for companies that will adopt such solutions. Final conclusion are expected together with the extensive analysis in Deliverable 5.4.

## 3.2 Project Business Models

The PRESERVE project provides different technologies which can be directly reused in FOTs or in industrial projects. This section aims at providing more information regarding the business model of PRESERVE results. Individual exploitation is not in the scope of this section but will be detailed in the deliverable D6.4.

The project envisioned different kind of business with the obtained results. Prior to the details of the exploitation possibilities, the selected business model framework is presented in the next subsections. Then, we will present our business vision.

### 3.2.1 Overview of the Business Model Framework

Figure 3.1 presents the Canvas framework[7] selected by the projects for describing our business models (BM). The model supports the most important aspect and allows us to avoid some missing parts in the business model. The blocks which are highlighted in red in Figure 3.1 corresponds to the core of a BM. In the business models presented in the next subsections, they covers these parts.

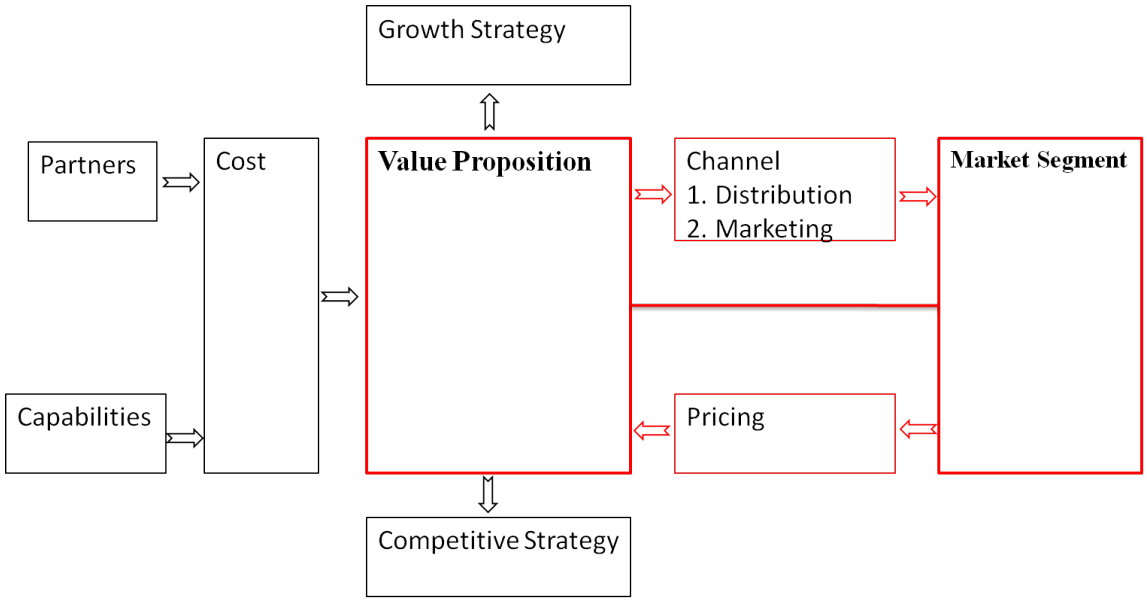



Figure 3.1: Template of the Business Modeling Framework

[7] see http://www.businessmodelgeneration.com/canvas

The following bullets describes the Canvas model:

- **Value Proposition** explains the value that the technology will bring to the customers in particular: (i) the offer to the users (how and why it addresses the need or job they need to do), (ii) how the users themselves would describe their benefit (articulate simple value propositions that the user can understand), and (iii) propose how to assess whether real value is created (e.g. users appreciating the product, using it in their life, saving money/time with it).
- **Market Segment** identifies the group of compagnies that will benefit from the value proposition. It is important to estimate how many they are now and how many they are expected to be in the future. Moreover, the market segment describes the motivation for the target markets to use the PRESERVE technologies.
- **Distribution Channel** describes how the value proposition will be delivered to the users. Multiple distribution channels can be selected. Finally, identifying the way to instrument the channels in order to get their performances.
- **Marketing Channel** aims to aware the final user about the value proposition. The way to communicate to the user and to receive their feedback has to be considered.
- **Pricing Model** has to be considered (e.g., fixed or variable prices, subscription). In PRESERVE, some building blocks are under open source licence. For this reason, some alternative sources of revenue has to be identified. For instance, the business model around consulting described in the next subsection is a way to get revenue with open source projects.
- **Competitive Strategy** aims at identifying the competitors of our value proposition.
- **Cost** is an analysis of the creating and delivering costs. In particular, the partnership has to be considered (i.e., other companies needed for building the value proposition) and also the capabilities (i.e., internal resources required to developed the value proposition).
- **Growth Strategy** allows the anticipation of the strategy.

### 3.2.2 Description of the Engineering Consulting Business Model

The first business model addressed by PRESERVE is around consultancy. For this business model, we mainly rely on PRESERVE building blocks such as PCOM and the ASIC. In this section, we will describe the value proposition, the market, the channels and the pricing models.

#### 3.2.2.1 Value Proposition

Around the PRESERVE building blocks, it is possible to sell different services:

- Consultancy on (i) how to integrate secure V2X communication in a project, (ii) security by design.
- Implementing new features in the VSS kit and porting to new platforms
- Providing support and maintenance

The VSS kit developed in PRESERVE will showcase our skills. The VSS kit offers a lot of flexibility:

- Compliance with several V2X certificate standards (IEEE 1609 and ETSI 103 097)
- Independent of the communication stack thanks to the convergence layer
- Open source and Proprietary licence versions
- Software or ASIC based low level cryptographic components
- Compliant with different platforms

#### 3.2.2.2 Channel

For maximizing the market awareness of our value proposition, we plan to use the following channels:

- Provide the Open source VSS Kit in a public repository
- Distribute factsheet and flyers during specific events like the ITS congress, the Electric Vehicle Symposium or ETSI events
- Contact current customers in the area (e.g., partners of FOT where PRESERVE was tested, and industrial projects)
- Meet potential customers during industrial forums
- Enhance the awareness through papers and educational

#### 3.2.2.3 Pricing

The pricing model for consultancy is based on subcontracts and by selling some ASICs.

#### 3.2.2.4 Market Segment

Currently, the project validates its results in a FOT and an hybrid FOT. Small and large FOT are still valid in the future. Providing consultancy around security is a requirement for FOT where security is out of the scope. Moreover, industrial R&D projects can also considered our value proposition. In particular, they can be awared by previous collaboration (e.g. during FOT or previous projects).

### 3.2.3 Description of the PKI Operating Business Model

The PKI is an important building block developed in the context of PRESERVE. In particular, one of our PKI is used as the C2C-CC pilot PKI implementation. Partners have acquired competence in operating a PKI. In this business model, we will describe the value proposition, the market segment, the channel and the pricing models.

#### 3.2.3.1 Value Proposition

- Features provided by the value proposition
  - Support IEEE 1609 and ETSI 103 097
  - Generation of long-term certificates and pseudonym certificates
  - Different interface to ITS communication device production line (UDP, HTML, webservice)
  - Communication protocol to support automated certificate request by ITS stations
  - Anonymity or Pseudonymity (pseudonymity if resolution is required, e.g. misbehavior detection)
- Advantages
  - Easily portable to new platforms since based on Java
  - Adaptable to other standards or interfaces
  - Conformance tests of certificate formats with ETSI had been performed
- Offers
  - Offer customized implementation to automotive OEMs, suppliers, road operators, and PKI operators
  - Offer operation of PKI as a service
  - Support and maintenance provided by ESCRYPT and Fraunhofer SIT
  - Implementation of new features or porting to new platforms

– Consulting on how to integrate the PKI into the production environment of automotive OEMs, suppliers, road operators, and PKI operators

#### 3.2.3.2 Channel

The following channels can be used:

- Offer the PKI as customized product or service to automotive OEMs, suppliers, road operators, and PKI operators
- Factsheet and flyers (distribution in some specific events like the ITS Congress, Electric Vehicle Symposium, ETSI events)
- Promote the software by the web through websites of ESCRYPT or Fraunhofer SIT

#### 3.2.3.3 Pricing

The price of the customized implementation, the PKI as a service, the support and the consulting has not be fixed yet.

#### 3.2.3.4 Market Segment

The projects plans to address the following markets:

- Productive operation of ITS (e.g., Automotive OEMs, suppliers, road operators, and PKI operators)
- FOTs with specific requirements that are not available by the C2C-CC Pilot PKI implementation

### 3.2.4 Possible Business Related to Certification

Certification is a critical point for automotive and ITS. The project is not directly involved in this area. However, some partners has competences in this domain through their involvement in C2C-CC and their expertise in the common criteria. This business model will be defined for the next version of the deliverable.

### 3.2.5 Conclusion on the Business Models

This section has presented potential business models for PRESERVE. Theses models will be refined in the next deliverable.

# 4 Misbehavior Detection and Prevention

## 4.1 Framework

In this section we describe a classification of state of the art misbehavior detection mechanisms, which was developed in cooperation with the university of Ulm. It is based on [4] and a survey article that is still in process of being submitted. At the university of Ulm, work is currently being done to develop a framework for misbehavior detection using subjective logic, a logic framework developed by Jøsang et al. We plan to implement the results of this work into the PRESERVE hardware tests.

### 4.1.1 Introduction

Vehicular ad-hoc networks (VANETs) are networks that are created by equipping vehicles with wireless transmission equipment. VANETs offer great potential to improve road safety and to provide information and entertainment applications for drivers and passengers. Due to the unique properties of VANETs, this type of network has attracted many researchers, including those in the domain of security. The security challenges in VANETs include the requirement for strong privacy, the computationally constrained environment, and the ephemeral nature of connectivity.

VANETs have a number of characteristics that require fundamentally new approaches for security, which differ from existing IT security requirements. In order to satisfy these requirements, misbehavior detection is a key aspect that needs to be addressed.

- **Safety-critical usage scenario.** VANETs are deployed in a scenario where failure or malfunction may have severe consequences, including massive financial loss or loss of lives, either through accidents or massive traffic disruptions. In this sense, VANETs are often considered critical infrastructures (CI) and the misbehavior detection mechanisms developed for VANETs may be deployed to CI in the future.
- **No clear security perimeter.** In VANETs, there is no clear boundary between insiders and outsiders. Instead, the logically and physically distributed nature of these networks leads to unclear security perimeters and possible insider attacks. VANETs are cooperatively formed by vehicles and road-side equipment, which are under distributed ownership and control, and it needs to be assumed that some of the vehicles are under full control of attackers. In addition, road-side equipment may be compromised by attackers.

- **Limited physical security.** As nodes in VANETs are often distributed in a potentially hostile environment, they may be subject to hijacking, analysis, and reprogramming by attackers. Due to cost constraints, the protection against such hijacking is often limited. This is similar to a Wireless Sensor Network (WSN) for environmental monitoring, where nodes may be scattered randomly in the environment. Due to the long lifetime of vehicles, similar challenges can be found in both VANETs and in-vehicle networks.
- **Sensor values as security assets.** The primary security assets in VANETs are the sensor values and the actuators controlled based on this input (or indirect responses produced by the driver). Spoofing and manipulation of sensor data are thus primary attack vectors. For instance, in a VANET that is used for detecting traffic jams, an attacker may want to suppress certain sensor readings that would indicate a traffic jam, or inject sensor values that indicate a traffic jam where none exists.

In summary, VANETs will likely attract attackers that try to manipulate sensed data and influence the resulting actions taken by the system. Such attackers may participate as regular network entities either because attackers can easily join the VANET or hijack already participating nodes. Once an attacker has entered the VANET, she can easily inject spoofed information into the VANET and trigger incorrect behavior. From the perspective of the VANET, this attacker can be seen as a misbehaving node that is sending incorrect data. In addition to information injection and manipulation, other attack types are conceivable, such as compromising routing efficiency by not forwarding information for other nodes. In this paper, we focus on detection of information manipulation. Note we cannot necessarily distinguish whether information manipulation is due to malicious intent or due to faulty hardware. However, from an information quality perspective, the resulting countermeasures should arguably often be the same.

Classical IT security mechanisms, like encryption, signatures, access control, (signature-based) intrusion detection systems, and so forth, are not suitable to thwart such insider attacks. Instead, we need security mechanisms that can identify misbehavior, identify the misbehaving nodes, and react either by filtering out the incorrect data or excluding the misbehaving node from further participation in the VANET. Research on security in VANETs has already developed several novel ideas for these tasks, many of which align with the goals of critical infrastructures. We discuss some of these in the next subsection.

### 4.1.2 State of the Art

Significant existing work on misbehavior detection in VANETs has already been done. We discuss several important results from related work, followed by a classification of these mechanisms into several orthogonal groups. Finally, we provide an overview of the solved and open challenges that we have identified for VANETs.

Two years later and as an example for a clearly behavior-based mechanism, Hortelano et al. [5] have evaluated the usefulness of so-called watchdogs, a concept well known from MANET [6] for VANETs. The core idea is that each vehicle acts as an observer of the forwarding

behavior of its neighbors. That is, each vehicle monitors the packets it receives, as well as all packets that are broadcast by neighboring vehicles. Because the routing mechanisms used in the network are known to each vehicle, it can predict which packets should be re-broadcast by neighboring vehicles. Hence, the ratio of packets that *should* be forwarded by neighboring vehicles versus the number of packets that are *actually* forwarded can be used to measure the protocol adherence of other vehicles. To accommodate packet collisions and noise on the wireless medium, the required number of re-broadcasts is lowered by a certain threshold, to reduce the number of false positives. The presented approach is independent of the actual message content and can be adopted to different routing mechanisms. Once the watchdog detects malicious behavior, it is logged in a local file, but reports are not forwarded to other vehicles or a centralized instance. Evaluation results show that it is difficult to set the malicious behavior detection threshold, a globally fixed parameter if their scheme, to a value that offers a good trade-off between attacker detection and false positives. Hence, dynamic adaptation of thresholds for watchdog mechanisms is necessary. In addition, we note this paper does not address privacy in their analysis, making the true suitability for VANETs unclear at best. Last, the authors note that their system is vulnerable to several attacks.

The previously discussed behavioral mechanisms focus mainly on detection of attacks on the routing layer, that is, on message dropping, alteration, and replay attacks. In contrast, Hamieh et al. [7] have described a detection mechanism for jamming attacks based on detecting patterns in radio interference. The assumption is that an attacker will intelligently jam the radio signal only during the time where honest vehicles transmit. This approach, known as selective jamming, is a common technique to avoid being easily detected due to constant jamming of the wireless channel. The proposed approach makes use of the fact that a selective jamming attacker will wait until regular transmissions occur until she jams the wireless medium. Hence, a correlation coefficient between correct reception time and time where errors occur is calculated. If the correlation is high, that is, if the medium is jammed most of the time when regular reception should occur, the medium is considered jammed. In order to achieve useful results, the authors took into account realistic reception and error probabilities as a baseline. Only if the correlation is unusually high, the medium can be considered jammed. The proposed method is interesting, because the correlation can be passively calculated with a simple formula, and because detection selective jamming is an important problem that is often neglected in security-related works.

Another similar approach has been presented by Hsiao et al. [8]: here, the main goal was to determine whether a claimed event has actually happened. In order to prevent possible attacks, the senders collect a number of witnesses for each possible event. For space efficiency purposes, $z$-smallest probabilistic counting is used, reducing the required amount of signatures that need to be attached to the message. The idea of $z$-smallest is that, given $n$ elements uniformly distributed between $0$ and $1$, the $z$-smallest element gives an approximation of $n$ by calculating $\frac{z}{c}$, where $c$ is the value of the $z$-smallest element. To protect against inflation, that is, attackers that try to increase the number of witnesses of an event, each vehicle signs a hash of its vehicle id, the event type, location segment, and time of the event. Only the $z$-smallest signatures are kept with the aggregate. The attacker can then not produce enough signatures on hashes that fall into the $z$-smallest values, because the hashes can be verified by the receivers. Therefore, an attacker cannot artificially increase

the result. Important to mention is that there is no deflation protection in this scheme; the attacker can reduce the amount of signatures attached to the message. The authors argued that an attacker will only try to produce fake events, such as a fake accident, and not try to hide events. Hiding events may also be achieved more easily by a powerful jamming signal. The next interesting concept has been presented by Raya et al. [9] in 2008. They proposed a mechanism to judge whether incoming messages are trustworthy by analyzing incoming traffic using different factors. The authors note their scheme is data-centric, because they use their mechanisms to evaluate confidence in messages. However, we note that while their proposals use this data, their focus lies not on individual mechanisms that process the data, but rather on the trust that can be developed on the basis of data-centric mechanisms, which is considered a trust-based approach in our classification. Even though the authors claimed that their mechanism is data-centric, the mechanisms used to determine the validity of messages are essentially based on trust that is developed for nodes, which is then used to decide whether a message is trustworthy. The authors used a combination of three different factors: default trustworthiness, based on the type of certificate (e.g., police cars); event- or task-specific trustworthiness, which matches the type of vehicle to the event; and dynamic trustworthiness, which captures message-specific things like proximity to the event. Once each factor is known, their output is combined and input to the decision logic. The output is then used to decide whether to trust a certain piece of information. The authors evaluated a number of different decision logic implementations, but stated that no single mechanism performs best in all simulated network configurations. However, the Dempster-Shafer inference [10, 11] was identified as the most promising technique. Besides their trust evaluation based on the Dempster-Shafer interference [9], Raya et al. [12] have also been among the first to present a system for locally evicting nodes, including the possibility to perform global revocation as a result using a mechanism called LEAVE. In their scheme vehicles collect accusations about a likely attacker until the number of reports passes a certain threshold. If enough accusations are collected, the accused vehicle is evicted temporarily. Once a vehicle possesses enough accusations, it can disseminate an aggregated message that contains the accusation, as well as a sufficiently high number of supporting signatures from other nodes. Vehicles receiving such an aggregated report can then directly ignore the accused vehicle. A core advantage of this approach compared to reputation systems is that the latency of the detection mechanism is much lower; reputation systems require time in order to build trust. Raya et al. [12] thus argued that local eviction is especially suitable for vehicular networks because of the low communication overhead and quick reaction time to attacks compared to global revocation. However, global revocation based on analysis of the collected local reports is foreseen as an orthogonal countermeasure against persistent attackers. A disadvantage of the scheme is that it may be vulnerable to Sybil attacks or privacy issues, depending on the type and implementation of the pseudonyms that are used.

The authors of [13] provide a detailed analysis of Sybil attack detection through analysis of physical layer properties. They assume that antennas, gains and transmission powers are fixed and known to all users of the VANET. However, they allow attackers to modify their transmission power. By applying signal models, they use the received signal strength to determine the approximate distance to the sender and apply this to verify the GPS po-

sition transmitted in each beacon message. They show the theoretically possible areas where an attacker can transmit to cause the receiver to observe the desired received signal strength, in order to correspond to the received signal strength. The authors also analyze the effect of using different antenna models (bi-directional and omni-directional) for the receiver. As the authors point out, they do not consider special propagation models or GPS errors. Lo & Tsai [14] have introduced a particular attack called the illusion attack, where the attacker injects false information into the VANET. To protect against this attack, the authors propose a plausibility validation network (PVN), which consists mainly of a checking module and a rule database, which allows information in new messages or provided by ones' own sensors to be verified. The rule database contains a set of rules that govern whether certain information should be considered valid or not, by analyzing the individual fields and verifying them against each other based on the rules provided by the rule database. This set of rules is dependent on message type. A message is valid if it passes all relevant verifications. The authors go on to provide a list of these rules in order to detect fake vehicles, which includes dropping of duplicate messages, that the location should be in range and plausible, the time stamp should be checked and the velocity should be plausible. The authors provide a formula for verification for each rule. When a message is considered valid, no rule has detected an attack, which means that either the appropriate rule does not exist yet, or the message is legitimate. As the authors only consider attackers manipulating sensors (i.e., the attacker does not have key material), this will be capable of detecting most attacks. However, our attacker model allows the attacker to generate arbitrary signed messages, which means that the attacker can generate messages to pass the (known) rule database. The authors of [15] and [16] take a different approach: they verify transmitted CAMs by analyzing the sequence of messages to find the trajectory of each vehicle. By tracking a vehicle using a Kalman filter, they can verify the location contained within each CAM, thereby allowing the detection and correction of falsified data in CAMs. This works, because the Kalman filter allows the accurate prediction of movement even under the influence of errors. As a result, the Kalman filter allows vehicles to locally link pseudonyms with high probability, and features adjustment for errors and new vehicles. By defeating pseudonyms in this way, vehicles can check that vehicles are transmitting valid messages. Their scheme explicitly does not distinguish between malicious and faulty nodes, instead aiming to detect any misbehavior. This work has been exploited by more recent work to verify message sequences and illustrates the trade-off between security and privacy. We note that the existence of Kalman filters does not imply that privacy is void – the Kalman filter only provides accurate estimates when actually following a vehicle (similar to physically following it by driving behind it).

Golle et al. [17] have presented the earliest example of data-centric detection, which checks for consistency between messages. When inconsistency is encountered, it uses attacker modeling to find possible explanations. The paper describes a framework to analyze the consistency of messages transmitted by different nodes. The model relies on four core assumptions: nodes can bind observations to received communication, they can uniquely identify neighboring vehicles (that is, detect Sybil attacks using physical properties), they can authenticate to one another, and finally, the network graph should always be connected. In case inconsistencies are found, Occam's Razor is applied, meaning that the explanation with the least amount of attackers best explains the conflicts found. As

an example of how their model works, the authors introduce two models that determine the correct locations of vehicles in the network. Although this is one of the earliest works on data-centric security mechanisms in VANETs, the outline of their mechanism is still used as a guideline for data-centric security and secure aggregation schemes. Unfortunately, the detection component of this work is not evaluated for feasibility, due to an assumption that neighbors can immediately exchange derived information. As the original authors note, assuming this kind of connectivity is not very realistic, as the amount of data exchanged in their scheme is quite high and available bandwidth in VANETs is limited.

Leinmüller et al. [18, 19] describe a position verification mechanism that bases on a number of different algorithms (also called sensors), each of which attempts to detect malicious or selfish behavior. The position verification mechanism determines a trust value for each vehicle, but the focus of the paper lies on the sensors. The authors propose sensors of based on either consistency or plausibility (called cooperative and autonomous sensors respectively in the paper); we discuss the consistency sensors here, and several plausibility sensors. The cooperative sensors are based on neighbor tables and position beacons to avoid the requirement of dedicated hardware. First, pro-active exchange of neighbor tables can include positions or only include logical links between nodes. In both cases, beacons are checked against received neighbor tables by comparing the claimed positions for a particular node in the beacon and the table. When the tables do not include positions directly, nodes can extrapolate information using the maximum transmission range. Second, reactive position requests can be used as a more bandwidth-efficient sensor. These requests are sent when an unknown vehicle $M$ is encountered; a vehicle knows the position of its neighbors, and selects a subset of them as either rejector or acceptor, based on whether the neighbor is in transmission range or not. It then sends its request to this subset of neighbors, asking for the position of $M$. Neighbors that do not know $M$ will respond with a corresponding message; others will respond with a position. The sender can then compare the responses with the expected responses. Both of these mechanisms rely on an honest majority, but are capable of dealing with noisy sensor data.

Footprint [20] improves on this idea: it is another example of a scheme that exploits central authorities by using similarity of trajectories, which are generated using signed messages by RSUs that a vehicle passes. In Footprint, these trajectories are cryptographically protected, and consist of special signatures, requested by the vehicle from the RSUs is has seen while driving. Footprint works by bounding the potential set of valid distinct trajectories an attacker can create. This bound is, in the worst case, the power set of trajectories, but can be limited in size using a test (which we do not discuss in detail here). The authors assume that real trajectories are sufficiently distinct; by forcing the attacker to obtain signatures through the RSUs, the bound is created based on the real path of the attacker. Then, when detecting Sybil attacks, all trajectories that are suspiciously similar are considered as coming from the same vehicle (referred to as a Sybil community). The authors use the trajectories for every message as an authentication mechanism, which allows any vehicle to compute the Sybil communities and avoid Sybil attacks. The signatures of RSUs are time-dependent an unpredictable, which means that location privacy is achieved against long term tracking.

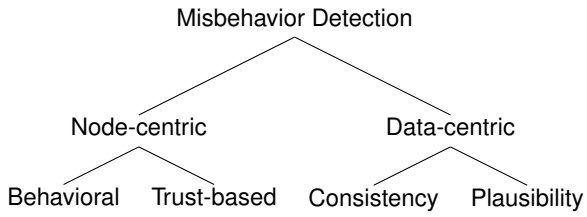



Figure 4.1: Taxonomy of misbehavior detection.

VARM is an approach that applies ideas from machine learning [21], which uses data mining techniques to perform misbehavior detection. The authors propose a data mining-based mechanism that dynamically derives association rules from received data using a data-structure called the Itemset-Tree. Association rules express correlations in a data set, and are a basic concept in data mining. By extracting these association rules, the node thus infers information from received messages that represent the expected behavior of senders. This knowledge can express hidden information that represents local road conditions, without the need to list all such scenarios and develop rules or models for them by hand. Their core drawback is that they may not generalize, because correlation does not imply causation. Another issue is that the paper does not extensively study the bandwidth requirements posed by their scheme, nor is latency or detection rate the main target of the study. We note that data mining is typically applied in scenarios where latency and computational resources are not an issue, and these techniques may not provide sufficient performance. Nevertheless, the application of data mining is a novel idea that can combine elements from both data-centric and behavioral misbehavior detection; this work provides a good starting point for applying data mining techniques to misbehavior detection. VARM is considered to be a consistency-based misbehavior detection mechanism because the authors specifically focus on temporal relationships between events received from many different vehicles, rather than verifying individual vehicles.

### 4.1.3 Classification

Golle et al. [17] have proposed a method to detect misbehavior as we defined it above in the context of VANETs. Instead of placing *trust* in nodes – as often done by classical cryptographic authentication mechanisms –, the proposed approach is to gain *confidence* in correctness of data by analyzing the local information base and deriving most probable explanations. During the following years, more research was done that proposes comparable misbehavior detection mechanisms for VANETs. Examples of these include [5, 9, 22–25], and [26].

There are fundamentally different approaches to misbehavior detection that can be used for a categorization of different mechanisms as shown in Figure 4.1. We first give a brief overview of each type, before discussing different types in detail.

A first distinction is whether mechanisms focus on data values contained in messages or on the node sending the messages. *Node-centric* mechanisms require authentication mechanisms to reliably distinguish between different nodes. Many systems achieve this

by assuming a trusted third party like a PKI that issues credentials, which are then used to authenticate messages and the corresponding information, using a security mechanism like digital signatures. Node-centric mechanisms can further be divided into *behavioral* and *trust-based* mechanisms. *Behavioral* mechanisms inspect a node's observable behavior (but not the information it is sending) and try to derive a metric that identifies how well a node behaves. For instance, a behavioral mechanism may inspect rates at which a neighboring node sends packets and decide whether a node significantly exceeds a "normal rate," which would then be considered as misbehavior. On the other hand, *trust-based* mechanisms inspect the past and present behavior of a node and use this to derive a probability for future misbehavior. The assumption is that a node who behaved correctly in the past is more likely to behave correctly in the future. Essentially, this boils down to some form of reputation management scheme where correct behavior increases the reputation while misbehavior reduces it. These mechanisms are commonly used for reporting and local revocation of nodes in a VANET, for example through LEAVE [12].

In contrast to those node-centric mechanisms, the second major category, namely *data-centric* misbehavior detection, subsumes all mechanisms that directly inspect the disseminated information to detect potential misbehavior. While data-centric mechanisms do not primarily care about the identities of individual nodes, they often still require some form of linking between messages to be able to reliably distinguish between different hosts. However, these mechanisms do not depend on the linkability of messages, which makes them highly valuable for the detection of Sybil attacks. Sybil attacks are a type of attack where a node replicates itself arbitrarily to undermine the honest majority assumption. Due to the strong privacy requirements in VANETs, which make linkage between different messages from the same sender more difficult, concerns for Sybil attacks are particularly relevant. In response to this, many VANET researchers have developed novel schemes to perform data-centric misbehavior detection; these can be divided further into *consistency* and *plausibility* mechanisms. Of these two types, *consistency* mechanisms rely more strongly on protection against Sybil attacks. The purpose of consistency mechanisms is to compare measurements from different entities to detect and, where possible, resolve conflicts between these measurements. For instance, in a VANET, a single vehicle could report a severe traffic jam while other vehicles report free flow of traffic. A consistency-based mechanism would use such information to conclude that there is likely no traffic jam and that the single vehicle may have misbehaved or be faulty. Finally, *plausibility* checking mechanisms are all mechanisms that have some implicit or explicit model of the real world and check whether incoming information is plausible within this model. For instance, in VANETs, speed reports of 700 km/h are not very plausible and may be filtered out. However, plausibility should be applied with caution in VANETs, as part of the focus of such networks is to detect outliers that indicate important, but rare, events, such as collisions between vehicles.

#### 4.1.3.1 Behavioral

Behavioral mechanisms are focused on the behavior of a particular node. This mainly concerns packet headers and meta-information like message frequency. Behavioral schemes

in VANETs typically focus on identifying nodes which send messages too frequently or nodes which modify the message content in a way that does not adhere to protocol standards. As these attacks are not fundamentally different from attacks that some classes of network intrusion detection mechanisms aim at, there are not many VANET-specific schemes available. Behavioral mechanisms are especially popular to protect networks where routing attacks and fairness play an important role, such as MANET; some of these misbehavior detection mechanisms have been adapted to work in VANET scenarios.

Before discussing mechanisms specifically designed for VANETs, we take a brief historical perspective and discuss a seminal work developed for MANET [6]. [6] introduces two tools for misbehavior detection: the Watchdog and the Pathrater, which they evaluate for the multi-hop routing mechanism called DSR. The essence of the watchdog mechanism is that each node that participates in routing monitors the network after forwarding a packet to a next hop. This node can then overhear whether the next hop forwards the packet or not, and therefore establish whether it is correctly behaving as defined by the protocol (in this case, DSR). Because a lossy channel might cause transmissions to be lost, a Watchdog should be configured with a threshold before it detects a node as malicious. Challenges for this mechanism include loss or collision on the channel, as well as false reports generated by malicious or colluding nodes. The second tool from [6] is Pathrater, which uses the watchdog results to rate the different network paths, so that the routing mechanism can select the best path even under the influence of attackers.

There are many types of behavioral mechanisms, including routing-oriented mechanisms that operate in a fashion similar to multi-hop routing in MANETs, as well as applications of machine learning algorithms to analyze which routes are of good quality and jamming detection mechanisms.

#### 4.1.3.2 Trust-based

Similar to behavioral mechanisms, many trust-based mechanisms for VANETs are rooted in mechanisms that were developed for MANET. These partially evolved from mechanisms such as the Watchdog [6] (discussed in the previous section), which provide metrics to establish the trustworthiness of a node. To aggregate this trust, distribute it among nodes, and provide it to a back-end system, a mechanism is required that not only filters malicious nodes as quickly and efficiently as possible, but also prevent attacks on the mechanism itself. For example, the Pathrater [6] tool aggregates the Watchdog results, but it may be attacked through Sybil attacks (as the authors also discuss). Core issues for trust-based mechanisms are Sybil attacks on the one hand, and high mobility and brief connectivity on the other. These challenges are much stronger in VANETs, as the connectivity between vehicles is sporadic, and privacy requirements lead to a vehicle being allowed to use multiple identities.

Trust-based mechanisms allow the participating nodes to vote on the correctness of data, or the trustworthiness of other nodes. Therefore most of these schemes employ some method of voting or agreement among nodes, typically relying on an honest majority. In the past, a number of common schemes have been surveyed in [27], which will be

taken into account in the following. The main focus are safety applications where vehicles broadcast messages to warn other vehicles about events like dangerous road conditions or accidents. Problems arise when misbehaving vehicles claim false events, and, hence, vehicles receive conflicting information about specific parts of the road.

Trust-based mechanisms include voting-based event validation, which allows different nodes to vote on the correctness of an event, as well as decision logics and reputation systems that are applied either directly or with an additional voting mechanism to remove misbehaving nodes directly from the network. Finally, several mechanisms have been proposed that attempt to link pseudonyms locally by cryptographic means.

#### 4.1.3.3 Plausibility

Plausibility checks can be used to quickly and efficiently filter packets that are malicious. Typically simple instances of these mechanisms are assumed to exist by node-centric schemes in order to provide a way to determine trustworthiness of nodes. However, plausibility checks can also be used as a more advanced tool to determine a numeric plausibility value, rather than just filtering out bad packets. For example, one can analyze the speed or location of a vehicle over time, a receiver can identify its path and attempt to identify suspicious paths. Plausibility checks are often used to detect attacks that involve Sybil nodes, as such situations still require that the attacker transmits from approximately the same location, despite the usage of many different identities.

A wide variety of plausibility mechanisms exists, which varies from highly accurate models like Kalman filters up to very rough checks that work by directly verifying individual messages, based only on their internal consistency. Next, there exist many position verification mechanisms that allow receivers to determine the correctness of the position based on channel information. Finally, Ghosh et al. [28] have noted that one can use plausibility to perform post-event validation, i.e., validating an event by analyzing the behavior of the driver.

#### 4.1.3.4 Consistency

Consistency-based mechanisms look at sequences of packets from distinct vehicles. These mechanisms focus on detecting and resolving conflicting information to achieve an accurate representation of the real world scenario. They are often employed by secure aggregation mechanisms to combine information from several vehicles into aggregates and to deal with inaccuracies, which may occur when aggregation mechanisms are used.

Consistency has been an important development in VANET research, because it allows for the most complex data-centric verification mechanisms. The types of consistency that exist include the direct checking of messages against each other to determine whether they conflict, Sybil attack detection using support from infrastructure, and a variety of more centralized schemes that can include the use of data mining to detect potential attackers or deviating patterns.

### 4.1.4 Misbehavior Framework using Subjective Logic

Note that no single mechanism alone will likely provide a convincing misbehavior detection mechanism that detects all forms and types of misbehavior. Instead, mechanisms will likely be combined. For instance, consider the following as an example for a combined approach. First, a number of data-centric mechanisms work on the same knowledge base to jointly detect incorrect data. Results are then augmented using behavioral mechanisms that check whether nodes behave according to protocol specifications. All these mechanisms are then used as input to a node-centric reputation management system that determines whether nodes show long-term misbehavior. These misbehaving nodes can then be reported to a central authority, which can determine whether nodes should be removed from the network; meanwhile, the nodes can be revoked temporarily by the nodes that detected the misbehavior. In the case of VANETs, the latter is particularly important, as this provides protection against determined attackers that may not be discouraged by high fines.

Based on our categorization, we are currently preparing a broad literature study on misbehavior detection in both VANETs and other CPSs. Our goal is to identify general patterns for misbehavior that work across specific application domains and scenarios, and can be re-used for a generic misbehavior detection architecture. This will allow application of security mechanisms developed for VANETs to be applied to a broader spectrum of problems, and could lead to security mechanisms developed for other CPSs to be applied to VANETs, furthering the safety and security of both.

## 4.2 Experimental Analysis of Misbehavior Detection and Prevention

### 4.2.1 Introduction

Whereas the V2X communication is an enrichment for the road traffic, an attacker that distributes bogus information may neutralize the positive effect, or worse, reduce traffic efficiency and safety. Hence, the exclusion of external attackers from the network by applying cryptographic security mechanism is very important. However, internal attackers with access to valid credentials or vehicular on-board systems are still a risk due to the decentralized character of the network and its inconsistent implementations of on-board architectures [29]. In the example of the EEBL application, an internal attacker would send a fake – but authenticated – emergency braking warning, which will generate erratic driving behavior and jeopardize the safety of neighboring vehicles. Therefore, misbehavior detection and prevention mechanisms, capable of filtering out wrong warnings, are required for a complete security solution. Still, to the best of our knowledge, no real-world experimental analyses on such attacks were done yet. We prove that internal attackers are a reality by exemplary implementing an EEBL attack. Furthermore, we show that the insufficient specifications of the on-board architecture of VANET nodes abet vulnerable implementations and therefore possibilities to create bogus messages without extracting

private keys. Then, we propose countermeasures and demonstrate the benefit of misbehavior detection systems using real vehicles.

The V2X communication system implementation for the test vehicles is based on the ETSI reference architecture [30]. The communication stack consists of several layers namely access layer, network & transport layer, facilities layer and application layer. Each layer maintains independently identifiers as well as location and time related information which might enable attacks. For external VANET communication two message types are considered. The Cooperative Awareness Message (CAM) is periodically broadcasted to single-hop neighbors. The Decentralized Environmental Notification Message (DENM) is sent on demand to inform neighboring nodes about unexpected events (e.g. congested areas or hazards on the road). The most important element of a CAM is the latest precise location of the sender in form of mobility data. This data set contains, beside other elements, the message generation timestamp, the current position as well as the heading, velocity, and acceleration of the sender. In order to protect the VANET against external attackers, certificates are used to digitally sign outgoing messages. Consequently, the authentication of the sender is ensured as well as the integrity of the signed message content.

### 4.2.2 Adversary Model

In the majority of the threat and vulnerability analyses, the distribution of bogus information, especially the cheating with position information, is identified as most threatening. Consequently, we focus on ways to forge positions using V2X communication systems that follow the ETSI reference architecture [30]. In Table 4.1 three types of attacker are distinguished. The Application Unit (AU) attacker is able to install malware or manipulate V2X application software. The Communication & Control Unit (CCU) attacker has full control over the communication router and on-board gateway, and a laptop attacker has full control over her/his own V2X communication facilities. The values for the metrics are

Table 4.1: Estimated effort and impact of position forging attacks

| **Metric** | **AU** | **CCU** | **Laptop** |
|---|---|---|---|
| Effort: Time | 2 | 3 | 4 |
| Effort: Knowledge | 2 | 4 | 4 |
| Effort: Access | 2 | 4 | 5 |
| $\sum$ **of efforts** | **6** | **11** | **13** |
| **Estimated impact** | **2** | **4** | **5** |
| **Effort / impact ratio** | **3** | **2.5** | **2.6** |

estimated based on related threat and vulnerability analyses [31, 32]. We use values in the range $[0, 5] \in \mathbb{N}$ whereby a high value means high effort or high impact. Since the time of attack preparation, the required attacker's knowledge and the required access to assets of the VANET are lowest for AU attackers, the AU attacks show the best effort-impact ratio. The payload (e.g. CAMs or DENMs) with forged content is generated by the AU attacker and sent out via CCU in the same way as it is done with unmodified payload.

Using well defined interfaces, the malware on the AU gets further data from the vehicle's internal CAN bus (i.e. location and mobility data), information about neighboring vehicles, and map data. Depending on the attacked applications of the target vehicles, it might be required for an attacker to suppress the dissemination of non-modified CAMs and replace them with forged ones. Since only the data fields of the application layer can be influenced, this kind of attacker can only affect the corresponding applications on a receiving AU. The mobility data of the other headers of a V2X packet are not affected as they are created by the CCU software.

We focus on the creation of ghost vehicles by adding forged mobility data to self generated messages. The ghost vehicle discussed here does not physically exist on the road, but V2X applications of receiving VANET nodes would think so. Due to navigation support and access to the V2X neighbor list, the malware can work automatically without manual interaction. It autonomously selects the location on the road where a ghost vehicle has probably the highest impact on neighbors.

#### 4.2.2.1 Misuse of the Emergency Electronic Brake Lights

We demonstrate the impact of an AU attacker by misusing the Emergency Electronic Brake Lights (EEBL) function as target application. The EEBL application is specified by the ETSI in the basic set of applications [33]. A strong braking vehicle, equipped with a V2X communication system, immediately broadcasts a DENM that informs the receivers about a panic braking action. After DENM reception, the EEBL application on single-hop neighbors calculates whether the braking vehicle is in its area of relevance. The relevance area is spanned in front of the receiver's vehicle with an angle $rel_\alpha = 90°$, and a length $rel_l = 400m$, (cf. Figure 4.2). If the DENM sender is inside the relevance area of the receiver, an information or warning is shown to the driver. On the AU of the attacker vehicle $A$ a malware is installed that analyzes the V2X neighborhood and automatically selects a victim $V$ as depicted in Figure 4.2. Then, a ghost vehicle $A_1$ is created in front of victim $V$

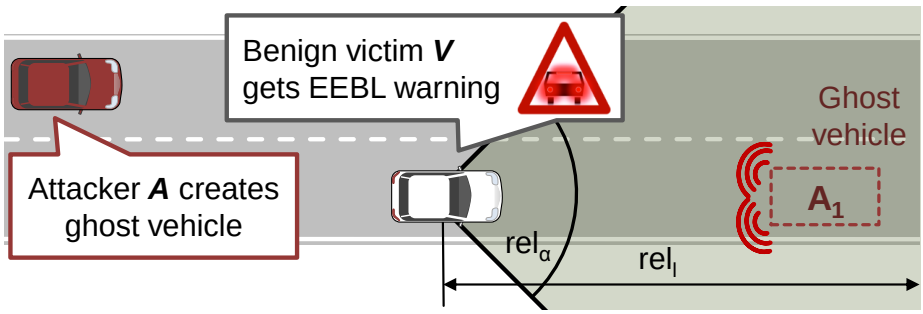



Figure 4.2: Simulation of a strong braking ghost vehicle $A_1$ created by attacker $A$

that pretends to drive in the same direction with a valid movement. After a lead time the attacker broadcasts an EEBL-DENM in the name of $A_1$ that informs about the fake braking action. The DENM and subsequent CAMs sent by the attacker contain mobility data with aligned positions and a negative acceleration value. Since $A_1$ is modeled in the relevant

safety area of $V$, the EEBL application of the victim displays a false driver warning. This may lead to an unexpected and possibly dangerous reaction of the driver.

#### 4.2.2.2 Setup and Implementation

For the experimental analysis three test cars has been used that were fully equipped with a V2X communication system. In our tests we modified only one vehicle by installing the malware application on the AU and deactivated the original CAM generation. All remaining components and functionalities on this attacker station has been left unchanged. The other two cars were not modified and served as victims.

Although different test variants were performed, we focus in the next section on the evaluation of a test scenario illustrated in Figure 4.2. In this scenario, an attacker is approaching the victim vehicle from behind, performs the attacks and falls back again afterwards.

#### 4.2.2.3 Evaluation of the EEBL attack

In the selected attack scenario an unmodified victim vehicle $V$ is driving with a constant speed of $\approx$ 14m/s on a straight road. The attack outcome on the unprotected receivers is

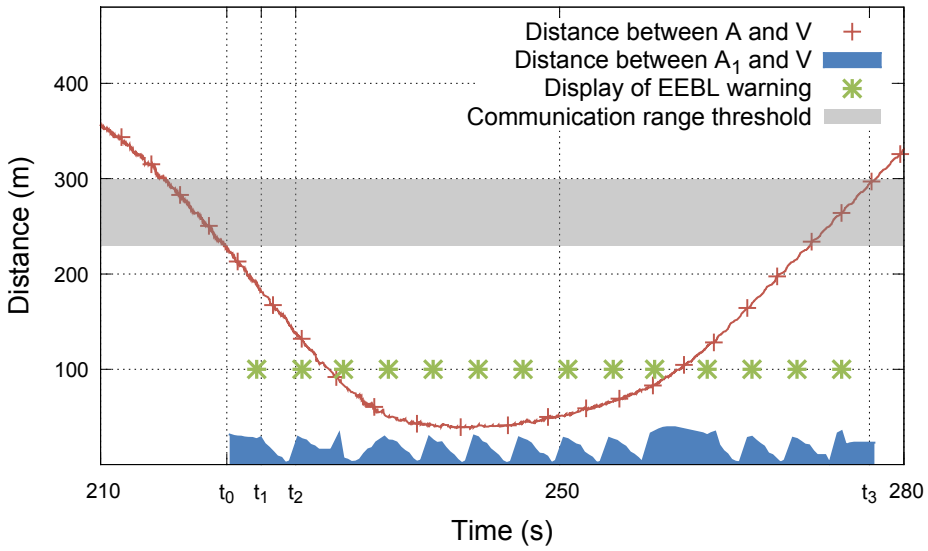



Figure 4.3: Attacker $A$ creates a braking ghost vehicle $A_1$ that provokes false driver warnings at receiver $V$. The victim $V$ is not running location data-based misbehavior detection and prevention mechanisms.

shown in Figure 4.3 with time and distance on the diagram axes. The diagram shows the attack between time $210$ and $280$.

At the beginning of this test, vehicle $A$ with the running malware drives 350 meters behind $V$ outside its communication range. The first curve shows the distance between attacker

$A$ and victim $V$ over the test time. After $A$ has approached $V$ and is located in single-hop communication range, the malware automatically detects the victim and executes the EEBL attack by creating the ghost vehicle $A_1$. Shown by the filled curve in Figure 4.3, the attacker creates CAMs for a ghost vehicle $A_1$ at time $t_0$ and waits 1 second before an EEBL warning is broadcasted in the name of $A_1$. At this point in time $A_1$ is placed approximately 30 meters in front of $V$. The ghost vehicle simulates an emergency braking action, decelerates and sends an EEBL-DENM at $t_1$ which is received and displayed by the victim $V$. Since the driver of $V$ is (intentionally) not reacting to the false warning the vehicle passes the position of the ghost vehicle a few seconds later. As soon as the malware detects that the ghost vehicle's position is passed by the victim, it places a new ghost vehicle in front of $V$ at time $t_2$ and starts another emergency braking attack. As a result, the victim $V$ gets a new warning at each iteration. This attack is repeated until $A$ leaves the single-hop communication range of the selected victim at time $t_3$. In all the performed tests the attacker $A$ was driving behind the victim $V$ in order to ensure that $A$ is not unintentionally in the EEBL relevance area of victim $V$.

### 4.2.3 Misbehavior Detection and Prevention

From an architectural perspective the different layers of the V2X communication stack (cf. ETSI reference architecture [30]) are independent from the data of other layers. In an optimal security solution each layer has to cryptographically protect its own data by adding a dedicated security header. In practice this strategy would enlarge the packet size dramatically and prevent a reliable high frequent broadcast communication. Consequently, a single security header per packet is considered. With the following countermeasures we aim to deal with all adversary categories discussed in Section 4.2.2. In this chapter the EEBL application is also used as a representative example.

#### 4.2.3.1 Sender Side Countermeasures

The main goal on sender side is to prevent the malicious modification of a VANET node and the dissemination of forged messages. The following countermeasures should be considered to harden the reference architecture.

- Firewalls between the interfaces of the communication stack shall prevent unauthorized message distribution.
- Generation time and generation location shall be contained in the security header of outgoing packets.
- Identifiers of the different headers shall be derived from the pseudonymous certificate or its certificate ID.
- Consistency and plausibility of location-based data contained in outgoing messages shall be checked.

- The software integrity of the AU and CCU shall be protected by secure boot mechanisms.

#### 4.2.3.2 Receiver Side Countermeasures

In order to detect and prevent the attacks on receiver side, the security subsystem of the VANET node should perform data consistency and plausibility checks in addition to cryptographic signature and certificate verifications. We propose to share meta data between the layers of the reference architecture. Especially, the identifiers and the location-based data contained in different headers of a received packet should be handed over through the layers of the V2X communication stack. The security subsystem shall collect all the IDs and location-based data with a consistency checker on the top most layer to compare the contents following a predefined consistency policy.

By applying the consistency checks on receiver side the attacker discussed in Section 4.2.2 can be detected and packets with inconsistent IDs and abnormal deviations in location data can be dropped. However, laptop attackers that are in possession of valid keys and certificates might be able to manipulate also the data of the cryptographically protected security header. Consequently, the application of location-based data plausibility checks [34–36] is approached in addition to consistency checks. In our practical experiments the plausibility checks are performed on application layer. The results of this type of misbehavior detection, applied on the same attack scenario as discussed in Section 4.2.2.3, are shown in Figure 4.4. In this figure the distance between the ghost vehicle $A_1$ and the victim $V$ is shown as well as all detected implausibilities. The plausibility checker on the AU analyzes the payload's mobility data of received CAMs and DENMs and creates a mobility tracker for every neighbor. If the attacker would create only a single EEBL-DENM without broadcasting CAMs, the plausibility checker of the single-hop receivers would evaluate the single DENM as implausible because the sender is not tracked and therefore unknown. Although the attacker broadcasts both, CAMs and DENMs he causes several plausibility violations that are detected as shown by the markers in Figure 4.4.

The sudden appearance of the ghost vehicle is detected when the attack is started for the first time (cf. time $t_0$ in Figure 4.4). The plausibility checker assumes that new neighbors usually appear at an outer margin area of the typical communication range. A second plausibility check detects position jumps of the ghost vehicle every time $A_1$ jumps to a new position in front of the victim, cf. time $t_2$ in Figure 4.4. Only abrupt jumps larger than 6 meters are considered as inconsistency as shown by the distance curve in Figure 4.4. A third plausibility check detects position overlaps of $A_1$ and $V$. Using the vehicle's position and dimensions, given by the CAMs, inconsistencies between the claimed area occupations of neighboring vehicles are detected [34].

Applying the proposed consistency and plausibility checks based on linked identifiers and mobility data no fake emergency braking warning is displayed to the driver of the victim vehicle $V$.

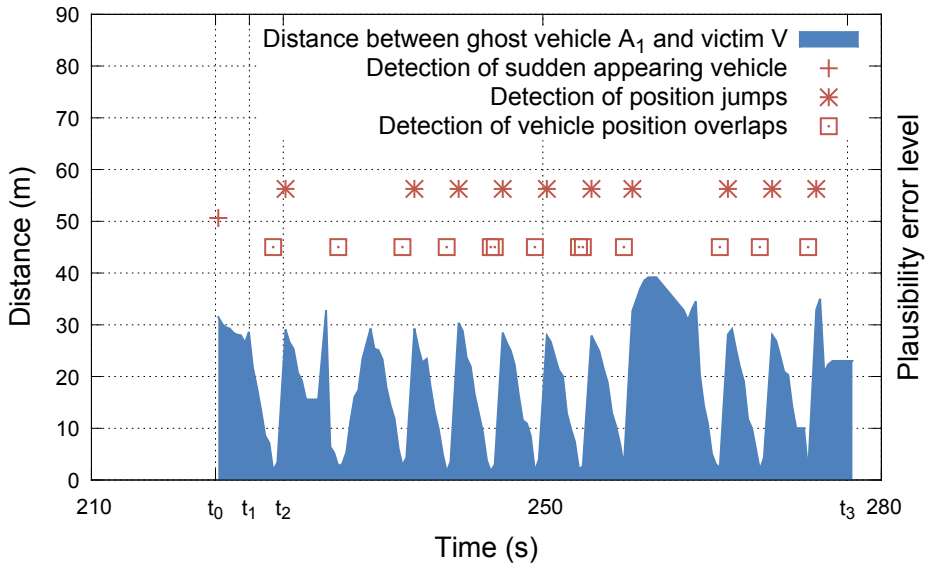



Figure 4.4: Location data plausibility check on receiver $V$ detecting the ghost vehicle $A_1$ that is generated by attacker $A$

### 4.2.4 Conclusion

Based on the implementation of an exemplary internal attack countermeasures are proposed to detect and prevent related attacks. Without misbehavior detection in place, malicious attackers can misuse the layered structure of a V2X communication architecture. In order to demonstrate the practical relevance and applicability of our countermeasures we installed an application layer malware that maliciously generates messages. With appropriate countermeasures applied on sender and receiver side misbehavior can be detected. Consequently false driver warnings can be prevented assuming an attacker on application and facilities layer. The experiments with three fully equipped vehicles further show that even ghost vehicles, generated maliciously by unrestricted laptop attackers, can be detected with mobility data plausibility checks. Based on observed implausibilities, attackers can be identified and excluded from the VANET on a long-term basis.

# 5 Privacy

## 5.1 Impact of Privacy on Intersection Collision Avoidance systems

The privacy protection schemes are not without consequences for safety applications. Such applications make decisions (e.g. warning drivers of an upcoming danger) based on their current estimation of the state of the real world, and this representation is created from the information contained in beacons received from other vehicles. Therefore, interruptions in the transmission of information will impact the decision-making process. If a silent period is scheduled to start at a safety-critical moment, it could result in safety systems not intervening when they should have, namely a "missed intervention". From a user and safety perspective, this is not acceptable.

In the paper [**?**], we address this issue and evaluate the impact of pseudonym change strategies on V2X-based collision avoidance systems. In particular we focus on Intersection Collision Avoidance (ICA) systems. This choice is motivated by the considerable potential of V2X-based safety applications to reduce the number of crashes at road intersections, compared to standalone safety systems. Indeed, a major issue for safety applications at road intersections is the potential occlusion of part of the scene due to the geometry of the intersection, the presence of obstacles like trees, buildings, etc. Some of the other vehicles can be detected by on-board exteroceptive sensors such as cameras, radars, or lidars, but others will be occluded or simply be beyond the field of view of the sensors. V2X communications do not suffer from this limitation and the hope is that this will help reducing the number of intersection-related accidents, which currently represent 40 to 50 percent of road accidents in most countries [37, 38].

### 5.1.1 Simulating privacy strategies

We simulate 3 different privacy protection strategies, described below.

The "*Fixed ID*" strategy assigns a fixed pseudonym to a vehicle for the entire duration of a trip (i.e. a new pseudonym is assigned to the vehicle every time it starts). Testing this case will give us a reference for how well the collision avoidance system performs when there is no pseudonym change and no silent period during a trip.

The "*Baseline*" strategy follows the recommendations of the SAE J2735 standard [39]. Pseudonyms are changed every $T_{chg} = 120$ seconds and are followed by a silent period of random duration $T_{sil}$ comprised between 0 and 13 seconds. Even if silent periods of duration shorter than 3 seconds are not considered in [39], we include them in our tests in order to analyze the impact of the silent period duration on the safety system.

The "*Adaptive*" strategy is a modified version of the *Baseline* strategy where the risk of the situation is taken into account to decide whether or not vehicle $i$ should be allowed to change pseudonym at time $t$. It relies on the computation of the probability $P(\textit{safety\_guaranteed}_{i,t})$, where the binary variable $\textit{safety\_guaranteed}_{i,t} \in \{0, 1\}$ corresponds to the current ability of the collision avoidance system to keep vehicle $i$ on a collision-free trajectory. A pseudonym change at time $t$ with a silent period of duration $T_{sil}$ is authorized if and only if:

$$P(\textit{safety\_guaranteed}_{i,t+T_{sil}}) \geq P(\textit{safety\_guaranteed}_{i,t}) \tag{5.1}$$

The idea here is to authorize a pseudonym change and silent period only if it will not affect the performance of the safety application. The computation of the terms in Eq. 5.1 will be detailed in the next section, after the description of the collision avoidance system.

By comparing the impact of these three privacy strategies on a collision avoidance application, we will be able to assess whether the standard "pseudonym change + silent period" strategy, here named *Baseline* strategy, affects the safety performance of the ICA system. It is also expected that the results will show whether the addition of a simple metric such as Eq. 5.1 is enough to prevent a loss of safety performance while providing some privacy protection.

### 5.1.2 V2X-based collision avoidance system

Several ICA systems have been proposed in the past which rely on V2X communications, e.g. [40–42]. The system used in this work is based on our previous work [42] where we proposed to evaluate the risk of a situation by estimating and comparing the intentions of the different drivers in the intersection area. The advantage of this approach is that it takes into account the dependencies between the motion of the different vehicles, which leads to a better assessment of the intentions of the drivers [43]. The approach was tested both in simulation [42] and in field experiments [43]. A brief description of the method is provided below.

#### 5.1.2.1 Probabilistic motion model

The joint motion of vehicles in a traffic scene is modeled by a Dynamic Bayesian Network (DBN) using four categories of variables:

- $I_{i,t}$ represents the maneuver being performed by vehicle $i$ at time $t$ (e.g. turn left, stop). We call it $I$ as in "Intention", since the maneuver performed by a vehicle reflects the *intended maneuver* of the driver.
- $E_{i,t}$ represents the maneuver that vehicle $i$ is expected to perform at time $t$ according to the traffic laws (e.g. turn left, stop). We call it $E$ as in "Expectation", since it represents the *expected maneuver*.
- $\Phi_{i,t}$ represents the *physical state* of vehicle $i$ at time $t$ (e.g. position, speed).
- $Z_{i,t}$ represents the *measurements* available about vehicle $i$ at time $t$. They often correspond to a noisy version of a subset of the *physical state* variables.

$I_{i,t}$, $E_{i,t}$, and $\Phi_{i,t}$ are hidden variables, while $Z_{i,t}$ is observed. For more clarity in the equations, in the remaining of this paper factored stated will be used to represent the conjunction of variables for the $N$ vehicles in the scene, e.g. $Z_t \triangleq (Z_{1,t}...Z_{N,t})$.

The proposed joint distribution of the DBN over all the vehicles is as follows [42]:

$$\begin{aligned} &P(E_{0:t_{end}} I_{0:t_{end}} \Phi_{0:t_{end}} Z_{0:t_{end}}) = P(E_0 I_0 \Phi_0 Z_0) \\ &\times \prod_{t=1}^{t_{end}} \times \prod_{i=1}^{N} [P(E_{i,t}|I_{t-1}\Phi_{t-1}) \times P(I_{i,t}|I_{i,t-1}E_{i,t}) \\ &\times P(\Phi_{i,t}|\Phi_{i,t-1}I_{i,t}) \times P(Z_{i,t}|\Phi_{i,t})] \end{aligned} \tag{5.2}$$

which corresponds to a classic Markov state-space model linking $I_{i,t}$, $\Phi_{i,t}$, and $Z_{i,t}$, augmented by the *expected maneuver* $E_{i,t}$ which is derived from the previous situational context $(I_{t-1}\Phi_{t-1})$ and has an influence on the intended maneuver $I_{i,t}$. For the interested reader more details about this model can be found in the previously published papers describing this DBN [42, 43].

#### 5.1.2.2 Bayesian inference for risk estimation

Inference on variables in the DBN described above is performed using a particle filter, which means that at each timestep the probability density function of the hidden variables $I_t$, $E_t$, and $\Phi_t$ is approximated by a set of weighted samples called particles. The set of $K$ particles at time $t$ is denoted:

$$\{H_{k,t}, w_{k,t}\}_{k=1:K} \tag{5.3}$$

with $H_{k,t}$ the state of particle $k$ at time $t$, and $w_{k,t}$ the weight of particle $k$ at time $t$.

The risk estimation algorithm proposed in [42] exploits the fact that 90% of road accidents are caused by driver error [44]. The probability of a collision in the future is computed as the probability that the intentions of drivers differ from what is expected of them:

$$P(\exists i \in N : \ I_{i,t} \neq E_{i,t}|Z_{0:t}) \tag{5.4}$$

Using the particle filter, this inference can be performed by summing up the weights of the current particles which verify the condition $(\exists i \in N : I_{i,t} \neq E_{i,t})$.

#### 5.1.2.3 Autonomous emergency braking

The collision avoidance application proposed in [42] triggers autonomous emergency braking if and only if the probability of a collision is higher than a threshold, i.e. iff:

$$P(\exists i \in N : I_{i,t} \neq E_{i,t} | Z_{0:t}) > \gamma \tag{5.5}$$

The threshold $\gamma$ was set after a precision / recall analysis [42]. The application runs in real-time on a dedicated dual core 2.26 GHz processor PC with 400 particles for the filter and with new observations $Z_t$ made available very 200 ms.

#### 5.1.2.4 Computation of $P(\textbf{\textit{safety\_guaranteed}})$

For the *Adaptive* privacy strategy introduced in Section 5.1.1, it is necessary to compute the probability $P(\textit{safety\_guaranteed})$. First of all we define the Time-To-Collision (TTC), and the Time-To-Stop (TTS). The TTC can be computed as the time that is left until a collision occurs if both vehicles involved in the collision continue on the same course and at the same speed [45]. The TTS corresponds to the time needed by a vehicle to reach a full stop after the ICA system intervenes, and can be computed as follows [37]:

$$TTS_{i,t} = \frac{s_{i,t}}{\delta} + T_{\textit{machine}} \tag{5.6}$$

with $s_{i,t}$ the speed of the vehicle $i$ at time $t$, $\delta = 7\,\text{m/s}^2$ the deceleration applied by the ICA system, and $T_{\textit{machine}} = 0.4\,\text{s}$ the average braking system response time [37].

The probability $P(\textit{safety\_guaranteed}_{i,t})$ that the collision avoidance system is currently able to keep the vehicle $i$ on a collision-free trajectory can be computed by summing up the weights of the current particles which verify the condition $(TTC_{i,t} > TTS_{i,t})$. The probability $P(\textit{safety\_guaranteed}_{i,t+T_{\textit{sil}}})$ that the collision avoidance system will able to keep the vehicle on a collision-free trajectory after a silent period of duration $T_{\textit{sil}}$ is computed by assuming constant speed during the silent period and summing up the weights of the current particles which verify the condition $(TTC_{i,t} - T_{\textit{sil}} > TTS_{i,t})$.

### 5.1.3 Results

#### 5.1.3.1 Evaluation metrics

In order to compare the three privacy strategies described in Section 5.1.1, we define metrics to evaluate both the level of privacy and the safety performance of the ICA application. The metrics are defined below.

**Rate of missed interventions** It is computed as $\frac{NM}{NC}$, with $NM$ the number of *collision* instances where the ICA system never intervened before the collision occurred and $NC$ the number of *collision* instances.

**Rate of avoided collisions** It is computed as $\frac{NA}{NC}$, with $NA$ the number of *collision* instances where the ICA system intervened and successfully avoided the collision and $NC$ the number of *collision* instances.

**Rate of failed interventions** It is computed as $\frac{NF}{NC}$, with $NF = NC - NM - NA$ the number of *collision* instances where the ICA system intervened before the collision occurred but was not able to avoid the collision and $NC$ the number of *collision* instances. Failed interventions, although not desirable, are still preferable to missed interventions. Indeed the system's intervention, even if triggered too late to avoid the accident, can be useful to mitigate the collision.

**Average privacy level** It is a unitless number computed over both *collision* and *no-collision* instances using the *user-centric location privacy model* introduced by Freudiger et al. [46]. In this model the privacy level of vehicle $i$ is defined based on the *location privacy loss function* $\beta_i(t, t_{chg,i}, T_{sil,i}) : (\mathbb{R}^+, \mathbb{R}^+, \mathbb{R}^+) \rightarrow \mathbb{R}^+$ where $t$ is the current time, $t_{chg,i} \leq t$ is the time of the last pseudonym change of vehicle $i$, and $T_{sil,i}$ is the duration of the silent period following the last pseudonym change. The privacy loss is set to zero after a change of pseudonym, remains zero for the duration of the silent period, then increases linearly with time according to a sensitivity parameter, $0 < \lambda < 1$ until it reaches a maximum $A_{max,i}(t_{chg,i})$. Thus, the privacy loss function is defined as follows:

$$\beta_i(t, t_{chg,i}, T_{sil,i}) = \begin{cases} 0 & \text{for } t_{chg,i} \leq t < t_{bro,i} \\ \lambda \cdot (t - t_{bro,i}) & \text{for } t_{bro,i} \leq t < t_{max,i} \\ A_{max,i}(t_{chg,i}) & \text{for } t_{max,i} \leq t \end{cases} \tag{5.7}$$

where $t_{bro,i} = t_{chg,i} + T_{sil,i}$ is the time at which the vehicle starts broadcasting again after a pseudonym change and a silent period, and $t_{max,i} = \frac{A_{max,i}(t_{chg,i})}{\lambda} + t_{bro,i}$ is the time when the function reaches the maximal privacy loss. Figure 5.1 illustrates the evolution of the function $\beta_i$ with time.

Using $\beta_i$, the privacy level $A_i(t)$ for vehicle $i$ at time $t$ is then computed as:

$$A_i(t) = A_{max,i}(t_{chg,i}) - \beta_i(t, t_{chg,i}, T_{sil,i}), t \geq t_{chg,i} \tag{5.8}$$

In practice it is generally assumed that $A_{max,i}(t_{chg,i}) = log_2(N)$, with $N$ the number of vehicles. Therefore in our case since $N = 2$ the privacy level computation simplifies to:

$$A_i(t) = 1 - \beta_i(t, t_{chg,i}, T_{sil,i}), t > t_{chg,i} \tag{5.9}$$

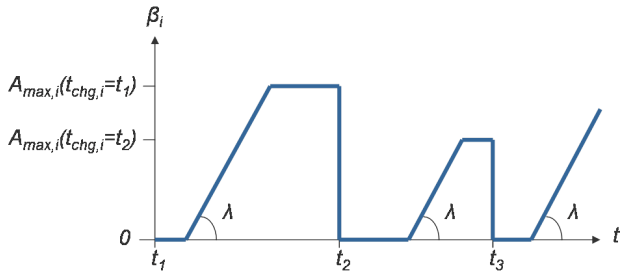



Figure 5.1: Location privacy loss function $\beta_i$ as a function of time. Vehicle $i$ changes pseudonym at times $t_{chg,i} = t_1, t_2, t_3$. Each pseudonym change is followed by a silent period of random duration where the privacy loss remains zero. At the end of the silent period the privacy loss increases linearly until it reaches a maximum $A_{max,i}(t_{chg,i})$.

Table 5.1: Comparison of the privacy strategies defined in Section 5.1.1 over all instances.

| | *Fixed ID* | *Baseline* | *Adaptive* |
|---|---|---|---|
| Missed interventions | 0.0% | 30.5% | 0.0% |
| Avoided collisions | 83.0% | 56.3% | 83.0% |
| Failed interventions | 17.0% | 13.2% | 17.0% |
| Average privacy level | 0.37 | 0.98 | 0.94 |

$\lambda$ models the tracking power of the adversary, therefore a higher value of $\lambda$ corresponds to a faster decrease of privacy loss. As advised in [47], we use $\lambda = 0.0005$, which means that the location privacy level is equal to zero after approximately 30 minutes without a pseudonym change. In other words, it assumes that after 30 minutes an attacker can track a vehicle and identify the driver.

#### 5.1.3.2 Comparative evaluation of privacy strategies

The rate of missed interventions, avoided collisions, failed interventions, and average privacy level are shown in Table 5.1 for the three tested privacy strategies.

The *Fixed ID* strategy never misses an intervention and is able to avoid 83% of the crashes. In 17% of the *collision* instances the ICA system intervened but triggering the emergency braking was not enough to avoid the collision. Typically, this happens when the OV slows down as if to stop when approaching the intersection and then accelerates at the last moment instead of stopping. The average privacy level obtained with no pseudonym changes is 0.37. Using Eq. 5.9, we find that this average privacy level is equivalent to the privacy level obtained after a 21 minutes long trip when the pseudonym stays fixed for the entire duration of the trip.

When applied on the same scenario instances, the *Baseline* strategy reaches an average privacy level of 0.98. Using Eq. 5.9, we find that this average privacy level is equivalent to the privacy level obtained after a 40 seconds long trip when the pseudonym stays fixed

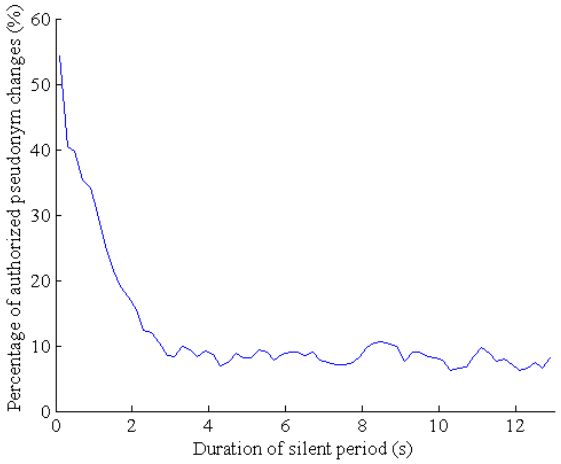



Figure 5.2: *Adaptive* strategy: Percentage of authorized pseudonym changes for the Other Vehicle as a function of the duration of the silent period.

for the entire duration of the trip. This improvement is brought by the introduction of pseudonym changes and silent periods, but is not without consequences on the performance of the ICA system. Indeed, the *Baseline* strategy has a high rate of missed interventions (30.5%) and a rate of avoided collisions which is 26.7% lower than the rate obtained by the *Fixed ID* strategy. The rate of failed interventions is lower for the *Baseline* strategy, but this is because some of the collisions that the *Fixed ID* strategy failed to avoid are now missed altogether by the *Baseline* strategy. The performance differences between the two strategies can be explained by the random occurrence of pseudonym changes and silent periods in the *Baseline* strategy. If a vehicle stops broadcasting information at a critical moment during *collision* instances, the ICA system may detect the danger too late.

The *Adaptive* strategy handles that issue by authorizing pseudonym changes only if they do not affect the safety application (see Section 5.1.1). The results show that adding this simple check is sufficient to restore the performance of the ICA system. As with the *Fixed ID* strategy, there are no missed interventions and 83% of collisions are avoided. The difference is that thanks to the pseudonym changes and silent periods, the privacy of users is much better protected: using Eq. 5.9, we find that a privacy level of 0.94 is equivalent to the privacy level obtained after a 2 minutes long trip when the pseudonym stays fixed for the entire duration of the trip.

### 5.1.4 Impact of the silent period

In this section we analyze further the results described above and investigate the decisions made by the *Adaptive* strategy to authorize or deny pseudonym changes with random silent periods. Figure 5.2 shows that the percentage of authorized pseudonym changes drops quickly from 55% to 15% as the silent period increases from 0.1 to 2 seconds. For longer silent periods, 10% of pseudonym changes are authorized on average.

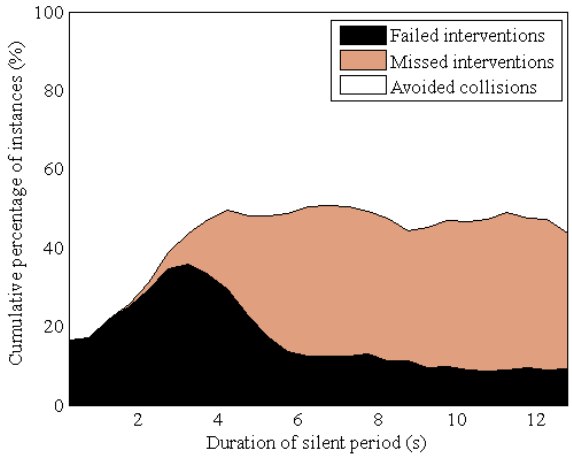



Figure 5.3: *Baseline* strategy: Percentage of missed interventions, avoided collisions, and failed interventions as a function of the duration of the silent period.

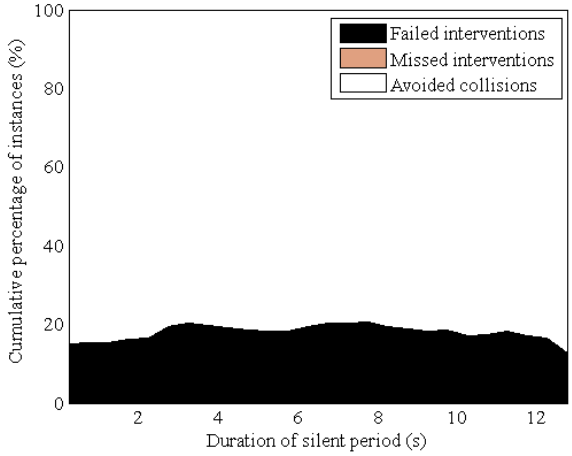



Figure 5.4: *Adaptive* strategy: Percentage of missed interventions, avoided collisions, and failed interventions as a function of the duration of the silent period.

Intuitively these observations can be explained by the fact that traffic at road intersections is highly dynamic: situations can become dangerous very quickly, and long silent periods can result in vehicles crossing intersections without broadcasting any information. This is incompatible with the objective of the ICA to ensure safety, and explains why the *Adaptive* strategy denies most pseudonym changes with silent periods longer than 2 seconds.

In order to verify this intuition we look at the distribution of missed interventions, avoided collisions, and failed interventions for different durations of the silent period. Figure 5.3 shows that introducing silent periods of duration smaller than 2 seconds leads to a slight increase of the rate of failed interventions for the *Baseline* strategy: 23% failed interventions on average against of 17% for the *Fixed ID* strategy. However these short silent periods do not result in missed interventions. For silent periods of duration comprised be-

tween 2 and 3 seconds, the rate of failed interventions keeps on rising and some missed interventions start occurring. For silent periods longer than 3 seconds, and as the duration increases, failed interventions are replaced by missed interventions. These observations confirm that silent periods longer than 2 seconds strongly affect the tested safety application, and explain why the *Adaptive* strategy rejects most of the pseudonym changes associated with long silent periods. By doing so, missed interventions are avoided and the rate of failed interventions is kept at the same level as the *Fixed ID* strategy, i.e. 17%, as shown in Figure 5.4.

### 5.1.5 Discussion

The main goal of this paper was to analyze the impact of privacy strategies on V2X safety applications, and the results presented above highlight the necessity of a joint design. That is, the requirements of safety applications should be taken into account when designing privacy strategies, and pseudonym change schemes should be accounted for when designing safety applications which rely on V2X communications. This collaboration is necessary in order to ensure that vehicular communications and safety applications do not neutralize each other, but instead, work together toward safer roads.

For example, the analysis conducted in this paper shows that the ICA application described in [42] requires silent periods to be shorter than two seconds in order to operate correctly in conjunction with the SAE J2735 standard (implemented here under the name "*Baseline* strategy"). The results also indicate that the addition of simple rules which authorize or not a pseudonym change depending on the context (implemented here under the name "*Adaptive* strategy") leads to major safety improvements compared to the SAE J2735 standard alone. Of course these results cannot be generalized to all V2X-based safety applications, since communication requirements may vary depending on the location (e.g. highway, rural road, intersection) and the application (e.g. collision avoidance, obstacle warning, emergency vehicle warning). We believe that studies similar to this one should be conducted in order to determine some "rules of thumb" around the design of V2X safety applications and privacy strategies to ensure that they work well together.

These studies could also explore new metrics to evaluate the safety and privacy levels. Indeed, the privacy loss function used in Eq. 7 only considers a linear increase. In order to represent a more realistic privacy loss, this function could for example consider the number of messages sent with the same pseudonym, the number of encountered neighbors (e.g, anonymity set size), or even the vehicle's mobility [48].

### 5.1.6 Conclusion and Future Work

Privacy is crucial in vehicular communications in order to ensure acceptance by users. To this end, the use of temporary pseudonyms has been proposed to provide a tradeoff between data privacy and security. However, this privacy mechanism is not without consequences for safety applications. In this paper we investigated the impact of pseudonym

change strategies on V2X-based Intersection Collision Avoidance (ICA) systems. We considered three privacy strategies and evaluated their performance both in terms of privacy and in terms of impact on the collision avoidance system. We found that the ICA system studied in this paper can operate correctly in conjunction with the SAE J2735 standard only if silent periods are shorter than two seconds. We also found that an "adaptive" strategy which takes into account the probability of a collision to decide whether a pseudonym change should be authorized or not provides a good compromise between ICA safety and privacy level. Future work should include similar investigations for other scenarios and other safety applications. It will be useful to consider a larger road network with more vehicles and various road topologies, so as to test more complex privacy strategies.

## 5.2 Privacy-Preserving Charging for eMobility

Mobility in the future has to become more eco-friendly. Especially, in urban scenarios the move towards electric mobility is already starting to become visible. This will bring along a fundamental transformation of the way how our transportation systems work, especially as electric vehicles will have to recharge much more often compared to the refueling of traditional cars. Charging of *Electric Vehicles (EVs)* is a central aspect of the electric vehicle introduction and a lot of attention is given to fast and widely available charging opportunities. Ideally, for the driver charging an EV will be as simple as parking – just park, plugin, and charging begins. Still, for charging control, authorization, and billing purposes, a lot of information has to be exchanged automatically between the EV and the *Electric Vehicle Supply Equipment (EVSE)*, also simply known as *charging station/spot (CS)*. Especially the multitude of different vehicle types and their electrical characteristics and requirements require a thorough setup of the EVSE. Moreover, frequent charging will also include frequent payments. The payment should be done without user-interaction for maximum convenience. In the context of ongoing international standardization efforts like ISO TC 22/IEC TC 69, it is manifesting that in the future certain roles of actors in the backend system will be concerned with security and privacy-related processes. Particularly, for the charging management of EVs, the draft standard ISO/IEC 15118-1 [49] defines actors and protocols to perform load management, billing and clearing, as well as certification. While security is already considered in the standards, privacy protection has not been investigated so far. In the paper [50], we present a detailed privacy analysis of ISO/IEC 15118 and propose modular privacy enhancements that lead to a fully privacy-preserving charging protocol for electric vehicles named POPCORN. In this section, we only focus on the POPCORN protocol and refer interested reader in Privacy Impact Assessment (PIA) to the full paper.

### 5.2.1 The POPCORN protocol

In the following the POPCORN protocol steps are explained in detail: contract establishment and installation of credentials, contract authentication, meter receipts, payment, and dispute resolution. Since the POPCORN protocol is based on the ISO/IEC 15118

standard, it uses the same message sequence, structure and general trust and security requirements, unless otherwise stated.

Phase 0 of the POPCORN protocol is only required when the EV user signs a new mobility contract. The phases 1-4 occur during every charging session. Phase 5 is only required in case of disputes.

**Phase 0: Mobility contract establishment** The contract establishment including the anonymous credential and group membership certificate installation is illustrated in Fig. 5.5. At vehicle production the OEM installs a Provisioning Certificate (also referred to as Bootstrap Certificate) in the vehicle (Step (a)). To charge using contract-based payment, the EV user signs a mobility contract with a mobility operator and registers the vehicle with the mobility operator (Step (b)). The mobility operator asks a global certificate authority to generate the anonymous credentials for the vehicle. The Idemix system allows the mobility operator to hide the contract attributes from the certificate authority, so that no private information about the user is revealed.

Before the first charging session using the mobility contract, the anonymous contract credentials and the group signature credentials have to be installed in the electric vehicle. The electric vehicle contacts the mobility operator for credential installation, for example, using the user's home Internet connection or the cellular network (Step (c)). The anonymous credentials including any other relevant contract attributes, e.g., special tariffs, are installed in the vehicle (Step (d)). That way, the electric vehicle also receives the certified public key of the mobility operator, which is used later to encrypt the SDRs. To obtain the group membership certificate, the vehicle generates a secret key for the group membership (Step (e)) and contacts the group manager, i.e., the dispute resolver, to obtain the group signing credentials (Step (f)+(g)). Now the electric vehicle can charge using the automated billing feature.

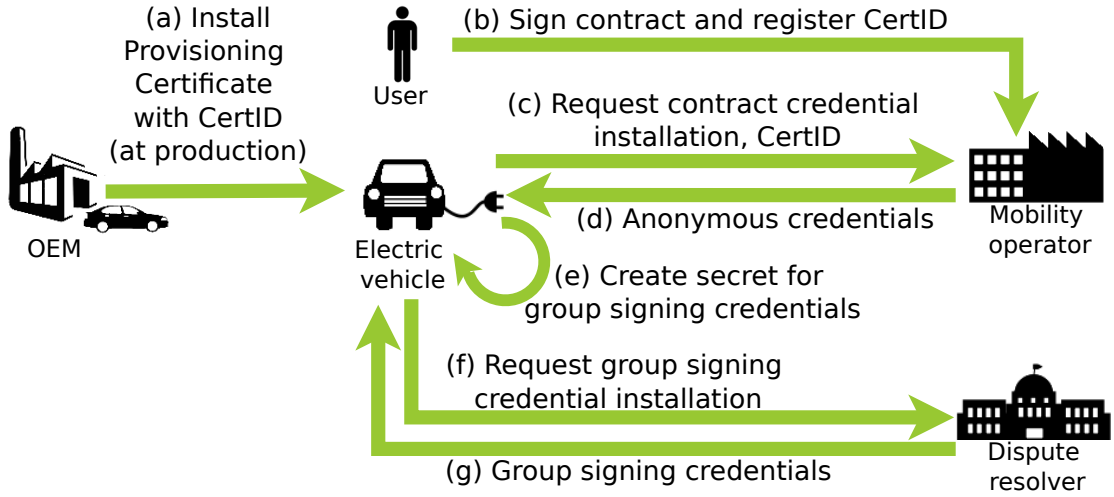



Figure 5.5: The POPCORN contract establishment.

Occasionally, the vehicle has to check for updates for its credentials. The anonymous credentials have a short lifetime in order to avoid having to use more complex revocation strategies. Similarly, the group membership has to be updated when vehicles leave the group. The charging station will refuse a signature and stop the charging session, if the vehicle has outdated group credentials. Most updates are non-interactive and can be downloaded by the vehicle when it has an online connection like a home access point or using an update method already defined by the ISO/IEC 15118 standard. If necessary, the mobility operator may contact the vehicle to inform it about a necessary update, e.g., to update the public key of the mobility operator.

The following POPCORN protocol phases 1-5 are depicted in Fig. 5.6.

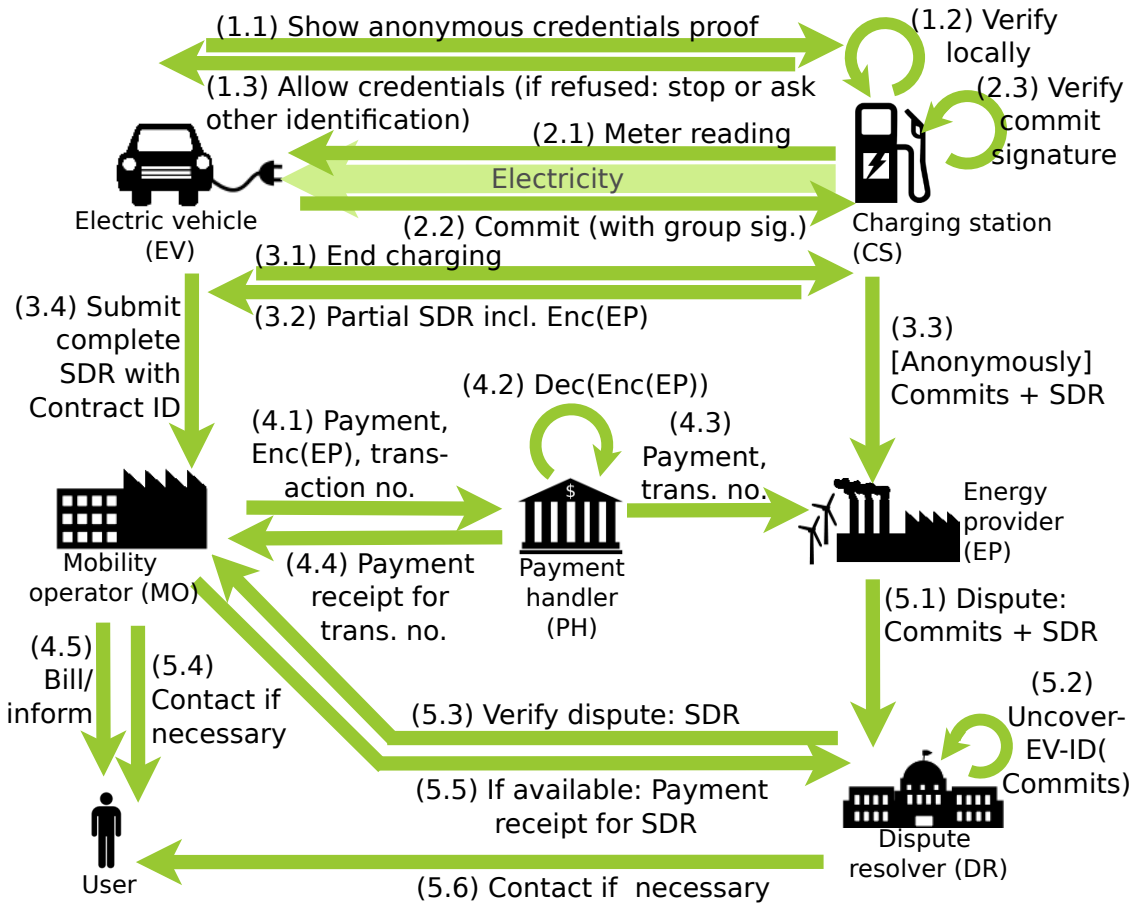



Figure 5.6: The POPCORN protocol for charging with automated payment.

**Phase 1: Contract authentication** When the electric vehicle is plugged into a charging station, the electric vehicle and charging station establish a communication link as defined in ISO/IEC 15118. The standard requires server-side TLS authentication to enable an authenticated and encrypted channel between the vehicle and the charging station. Also, the POPCORN protocol requires this. Next, for contract authentication, the electric

vehicle proves to the charging station that it has a valid contract using its anonymous credentials (Step (1.1)). To show that the contract is still valid the vehicle applies an attribute operation to compare the contract end date with the current date. The vehicle does not disclose any other contract attributes. The vehicle also has to prove that the anonymous credentials' validity period has not expired to show that the credentials have not been revoked. The charging station can locally verify the validity of the proof (Step (1.2)) and hence the charging contract (Step (1.3)). If the vehicle is eligible to special tariffs, the electric vehicle can use its credentials to prove this to the charging station by performing another attribute proof.

**Phase 2: Charging loop with meter receipts** During the charging loop, the charging station sends the current meter reading to the electric vehicle after some fixed amount of energy has been delivered (Step (2.1)). The electric vehicle then generates a group signature over the reading and sends the resulting payment commitment back (Step (2.2)). The charging station verifies the signature with the group's public key (Step (2.3)). If the signature is valid, the charging cycle continues, otherwise the charging station aborts the charging session.

At the end of the charging session, the charging station generates a partial SDR and sends it to the electric vehicle (Step (3.1)+(3.2)). Now the connection between the electric vehicle and the charging station is terminated. The charging station anonymously forwards, e.g., via a TOR network, the group-signed commitments and the partial SDR to the energy provider. The energy provider can link the charging session to an incoming payment using the transaction number contained in the SDR (Step (3.3)). Since the information is transferred anonymously, the energy provider cannot link the charging session to a specific charging station.

**Phase 3 and 4: SDR delivery and payment** The electric vehicle probabilistically encrypts its Contract ID and appends it to the partial SDR. The electric vehicle then signs the complete SDR. When the electric vehicle has access to an Internet connection, e.g., using the cellular network or the user's home Internet WLAN, the complete encrypted SDR is submitted to the mobility operator (Step (3.4)). The vehicle should not use the charging station's Internet service for this forwarding, as this may reveal the charging location based on the source IP address. In this case, the vehicle has to make use of a privacy proxy. The mobility operator verifies the signature and uncovers the Contract ID. The mobility operator now knows which user the bill belongs to and can inform and bill the user for the charging session (Step (4.3)). The mobility operator will, however, not learn the charging station or energy provider identities.

In order to complete the processing of the SDR the mobility operator sends the payment with the encrypted energy provider value and the transaction number (both contained in the SDR) to the payment handler (Step (4.1)). The payment handler decrypts the identity of the energy provider and forwards the payment and transaction number accordingly (Step (4.2)+(4.3)). Finally, the payment handler sends a receipt to the mobility operator, to confirm the payment (Step (4.4)).

**Phase 5: Dispute resolution** If the payment of a charging session does not arrive within the defined payment period, the energy provider can contact the dispute resolver with the group-signed meter readings and the partial SDR (Step (5.1)). The dispute resolver has access to the group's secret key and can uncover the vehicle's identity from the commits (Step (5.2)). As a first step, the dispute resolver contacts the mobility operator of the vehicle in questions and requests the payment receipt (Step (5.3)). The mobility operator has to check his records for the given SDR. If the mobility operator cannot send the matching receipt, the mobility operator has to fulfill the missing payment. In addition, the mobility operator may verify with his customer why no SDR was submitted for the charging session (Step (5.5)). The dispute is resolved when a valid receipt for the transaction in question is shown to the dispute resolver (Step (5.4)).

It should be noted that the above consideration assumed that energy provider and mobility operator are different legal entities. In case the energy provider is the same as the mobility operator, the protocol still ensures privacy. Any charging station anonymously sends the charging details to the energy provider, so that the combined MO-EP cannot deduce where the mobility operator was charged.

Table 5.2 summarizes the major modifications of POPCORN compared to ISO/IEC 15118.

### 5.2.2 Conclusion

In this work we have highlighted the privacy invasion that electric vehicle charging based on ISO/IEC 15118 may introduce. As our privacy impact assessment of this protocol has shown, drivers may unnecessarily reveal details about their whereabouts to charging station and mobility operators. Using our PIA results, we designed modular enhancements of the protocol based on state-of-the-art PETs, showing that PET technology allows to implement comfortable and fully functional Authentication, Authorization and Accounting (AAA) for eMobility and electric vehicle charging without sacrificing privacy. This claim was corroborated by a second PIA analysis and a prototype implementation.

By taking a modular approach to extend the original ISO/IEC 15118 protocol, POPCORN can even be introduced in a gradual way, if industry is not willing to initially introduce a dispute resolver or payment handler. Of course this goes at a reduced privacy protection. Still it would allow an immediate introduction of better privacy protection to the current protocols and infrastructures. We are in the process of submitting our POPCORN proposal to the respective ISO working group to discuss the potential for actual consideration in the standard.

We have the hope that our work will provide a significant contribution to the introduction of privacy-preserving and still functional and convenient electric vehicle charging infrastructures. At the same time, it provides a lesson how today's PETs in combination with thorough PIA can be used to build and deploy privacy-enhancing systems that introduce only modest additional effort but fully retain system functionality and security.

ISO/IEC 15118 will mainly affect the European market. In the U.S., SAE J2836/2847 will play a similar role. We plan to investigate the security protection of this protocol as a next step in our effort to bring privacy-preserving eMobility closer to reality.

Table 5.2: Comparison of ISO/IEC 15118 and POPCORN protocol.

| **Activity** | **ISO/IEC 15118** | **POPCORN** |
|---|---|---|
| Contract authentication | Certificate and Contract ID | Anonymous credentials with contract attributes, incl. Contract ID |
| Determine tariffs | Contract ID or attribute certificate | Anonymous credentials attributes |
| Contract establishment | Provisioning Certificate registered with mobility operator | Identical to ISO/IEC 15118 |
| Credential installation / update | EV obtains: Contract Cert. and ID, CRLs | EV obtains: Anonymous credentials, Contract ID, group membership certificate |
| Contract authentication | EV shows Contract Cert. and ID, CS verifies with backend | EV proves contract validity with anonymous credentials, CS verifies locally |
| Meter reading commitment | EV signings with its signing key, CS can verify signature, sent to EP | EV generates group signature, CS can verify signature, sent to EP |
| SDR delivery | CS generates and delivers SDR | CS generates partial SDR, EV appends extra values, signs and delivers SDR |
| Payment | MO reads SDR and pays EP | MO reads SDR and sends payment and encrypted receiver value to PH, PH decrypts receiver and forwards payment, PH sends receipt to MO |
| Dispute | EP uncovers EV identity from signature and contacts EV/MO | EP submits dispute with DR, DR verifies and uncovers EV identity from group signature, DR contacts MO/EV and obtains payment receipt to resolve dispute |

# 6 Identity Management

## 6.1 Pilot Public Key Infrastructure

The Pilot PKI of the Car-to-Car Communication Consortium is an infrastructure that is to be used for testing purposes in development processes.

The Pilot PKI consists at minimum of three different CAs following different roles. As shown in Figure 6.1, the Root CA (RCA) manages the root certificate and issues the Long-Term CA (LTCA) and the Pseudonym CA (PCA) on top of the PKI hierarchy. The

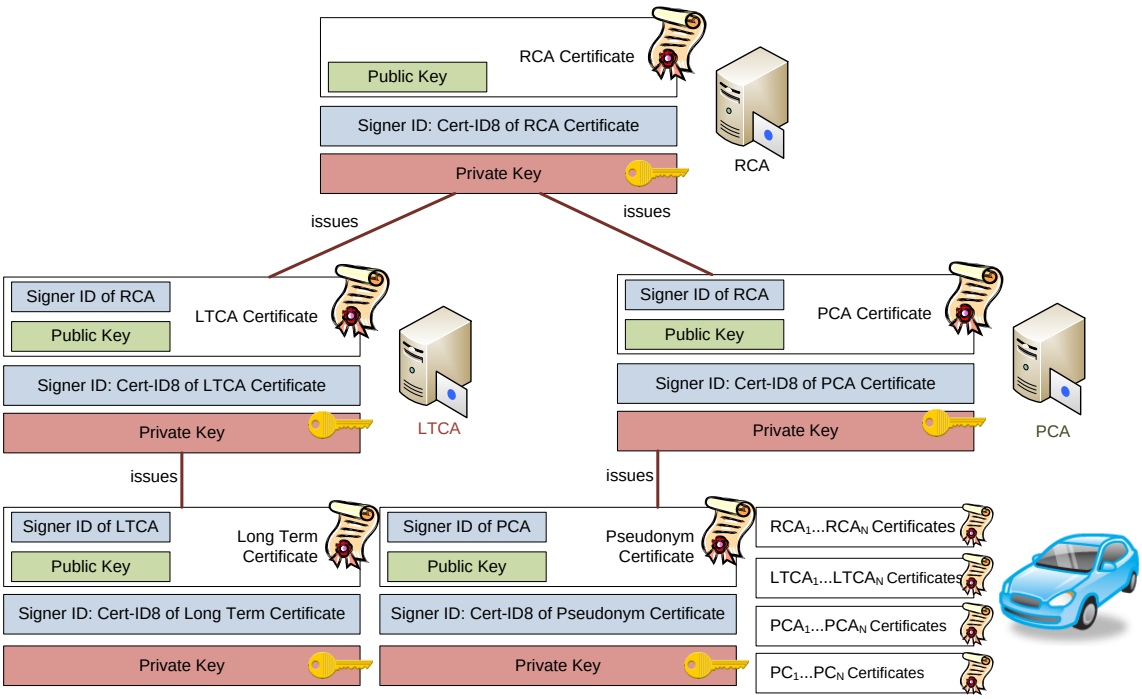



Figure 6.1: PKI hierarchy

RCA is the trust anchor in the PKI issuing certificates for LTCAs and PCAs. The PKI is designed to be limited to these two layers. An LTCA or a PCA is not permitted to issue additional intermediate CAs. The LTCA is responsible to manage registered ITS stations and issues Long-Term Certificates (LTC) that are used to request Pseudonym Certificates (PC). The ITS station requests the LTC from the LTCA and PCs from the PCA.

### 6.1.1 The Pilot PKI's components

#### 6.1.1.1 RCA

The Root CA is operated by Fraunhofer SIT.

**Interfaces** The RCA operates a webpage that can be accessed via Interne. The webpage provides the following services or information:

- Dashboard with general information of the CA (available without access restrictions)
- Download of Root Certificate as hex-encoded string (available without access restrictions)
- Download of CRL as hex-encoded string (available without access restrictions)
- Display of issued CA Certificates (user login required)
- Access to Logs (user login required)
- Performance statistics (user login required)

The RCA provides a SOAP webservice which can be used to download CA certificates or CRLs. The usage of this webservice does not require client authentication as the accessed data is public.

#### 6.1.1.2 LTCA

The LTCA is operated by Fraunhofer SIT.

**Interfaces** The LTCA operates a webpage that can be accessed via Internet. The webpage provides the following services or information:

- Dashboard with general information about the CA (available without access restrictions)
- Download of LTCA Certificate (available without access restrictions)
- Download of CRL (available without access restrictions)
- Registration of Authenticators (user login required)
- Display of Registered Authenticators (user login required)
- Deletion of Registered Authenticators (user login required)
- Registration of ITS Stations (user login required)
- Display of Registered ITS Stations (user login required)
- Deactivation of Registered ITS Stations (user login required)

- Re-activation of deactivated ITS Stations (user login required)
- Display of Issued Long-Term Certificates (user login required)
- Display Authenticated Pseudonym Intervals (user login required)
- Access to Logs (user login required)
- Performance statistics (user login required)

The LTCA provides two different SOAP webservices. One webservice does not require client authentication as the request messages contain the authentication information itself. It provides the following services:

- Download of LTCA certificate
- Request of CA configuration
- Request of Long-Term Certificate

A second webservice with client authentication can be used to register multiple ITS stations in one batch. Its specification can be accessed via the Internet.

**Registration of authenticators** Pilot PKI users can register and delete one or several different authenticators via LTCA webpage. The authenticators can be used to automatically register multiple ITS stations in one batch. The following information is required for a registration:

- Personal information ( **company, name, email**, telephone, address)
- X.509v3 certificate that is applied by the authenticator to use the web service of the LTCA in order to register the ITS stations. The email address in the X.509v3 certificate has to be the same as used in the personal information.
- The maximum **assurance level** that the authenticator is authorized to validate

Optionally, the following items may be defined

- A list of AID, AID_Priority, AID_Priority_SSP, AID_SSP according to ETSI TS 103 097 that the authenticator is authorized to validate with its digital signature
- Regional restrictions as defined in ETSI TS 103 097 that may limit the validity area of the authenticator

The authenticator has to use the fixed ID block for the 8 most significant bytes of the module id when it automatically registers ITS stations.

**How to register ITS stations** The registration of an ITS station requires the following data:

- **16-byte module id**: This id is unique per LTCA and identifies the ITS station's security module. The module id must contain the fixed ID block at the 8 most significant bytes. A method to securely link ITS station to its security module has not been defined in the context of the Pilot PKI and is yet to be defined.
- **Module authentication key**: The ITS station's security module contains a randomly generated ECC-P256 key pair. The public key must be transferred to the LTCA during the registration process. This key is later used to sign the request for a long-term certificate.
- **Assurance level**: The assurance level that the ITS station's security system has received at an independent certification according to a pre-defined protection profile. The protection profile is currently developed within the working group security.

Optionally, the following items may be defined

- AID, AID_Priority, AID_Priority_SSP, AID_SSP: See ETSI TS 103 097 for explanation of these information
- Regional restrictions as defined in ETSI TS 103 097

**Management of ITS stations** Deactivation and reactivation of ITS stations as well as displaying registered ITS stations can be performed using the LTCA's webpage.

**Requesting Long-Term certificates**

- Long-Term certificates can be requested using the appropriate webservice method with a request message of type LtcRequest of the Security Management Formats document.
- An ECC-P256 key pair must be generated first. The public must be put as subject attribute into the request message.
- The request message must be signed using the private module authentication key.
- The parameters of the request message shall be filled with those values that the requester wants to be written in the Long-Term certificate.

**Management of Long-Term Certificates and Authenticated Pseudonyms** Issued Long-Term Certificates are listed and can be searched by their certificate hashed ID on the LTCA's webpage. As the LTCA manages the number of Pseudonym Certificates that can be issued by the PCA, the pseudonym intervals related to every long-term certificates can be displayed on the LTCA's webpage. For testing purposes blocked pseudonym intervals can be cleared on the LTCA's webpage.

#### 6.1.1.3 PCA

The PCA is operated by ESCRYPT GmbH - Embedded Security.

**Interfaces** The PCA operates a webpage that can be accessed via Internet. The webpage provides the following services or information:

- General information about the CA (available without access restrictions)
- Download of PCA Certificate (available without access restrictions)

The PCA provides a webservice for requesting pseudonym certificates. The specifications can be accessed via the Internet.

**Requesting Pseudonym certificates**

- Pseudonym certificates can be requested using the appropriate webservice method with request message of type PcRequest of the Security Management Formats document.
- For each requested certificate, an ECC-P256 verification key pair must be generated. Optionally, encryption key pairs can be generated. The public keys must be put in the respective lists of the PcRequest message.
- The request message must be signed using the LTC private key.
- The parameters of the request message shall be filled with those values that the requester wants to be written in the Long-Term certificate.

#### 6.1.1.4 Client-Software

The client software has been developed by ESCRYPT GmbH - Embedded Security. The client software is a graphical user interface (GUI) that allows the user to test all automated processes for one single ITS station. It allows to run through the complete process from the registration of the ITS station to download of CA certificates, download of certificate revocation lists (CRL), requests of Long-Term certificates (LTC) to requesting pseudonym certificates (PC).

#### 6.1.1.5 Example Source code

The example source code of the Pilot PKI is part of the client software and is made available by ESCRYPT GmbH - Embedded Security. The source code contains a detailed documentation about the provided functions and the purpose of their parameters. It covers access to all interfaces of the Pilot PKI and allows for usage of these interfaces with their full parameter range. The following parts are the core Java classes that demonstrate the usage of the PKI:

- **Request Factory**: This Java class provides functions to create request messages in the format as specified in [51]. Since the arguments of these functions are Java objects, the process of how to compose such request messages becomes visible.
- **Webservice client**: The web service client consists of functions that take a request message and perform the communication with the respective CA. After receiving the response, it evaluates it and returns the expected object if the request has been successful
- **ITS station object**: This object represents an ITS station with all its required data. The methods of this object. The methods of this object implement the requests from the perspective of the ITS station using the functions of the request factory and the webservice client.

#### 6.1.1.6 Specification of message formats

The message formats that are used to communicate with the PKI are specified in a separate document [51].

#### 6.1.1.7 Certificate Policy (CP)

The Certificate Policy that is delivered with the Pilot PKI package lists regulations that have to be followed by the PKI operators and by users. Most relevant for users of the Pilot PKI are section 3 and 4 of this document, which describe the processes of ITS station registration and certificate requests and usage in detail. The CP can serve as a basis for a CP of a productive PKI and is to be improved or extended during the operation phases of the Pilot PKI.

#### 6.1.1.8 Certification Practice Statements (CPS)

In the Certification Practice Statements (CPS), the CA operators describe how the CA is operated. In particular, these CPS cover the regulations that are imposed upon them by the CP.

#### 6.1.1.9 Bugtracking and support

The bugtracking system of the Pilot PKI is operated by ESCRYPT GmbH - Embedded Security. With the registration at the Pilot PKI, each user gets access to the JIRA bugtracking and support system, which can be accessed via Internet. There are different types of issues that a user can create. Issues can be created by every user of the Pilot PKI. Users can also comment on existing issues making a discussion possible.

## 6.2 Conditional pseudonym resolution algorithm

### 6.2.1 Problem Statement

In VANETs the location privacy of drivers should be protected by using pseudonymous identifiers in messages that may change frequently to avoid linking of recorded identifiers. According to [52], a pseudonym is an identifier that is used by a subject instead of one of its real names. Due to privacy protection requirements initially unlinked short-term pseudonyms are required in ITS communications. It should not be possible to link these pseudonymous identifiers to their long-term identifier, neither by other vehicles nor by a single trusted third party. But in defined situations, conditional pseudonym resolution may be required due to different specific circumstances as motivated in the following examples. On the one hand, a Law Enforcement Agency (LEA) may need to get long-term vehicle information based on their initially non-public pseudonyms in order to identify involved drivers in case of a traffic accident. On the other hand, a Misbehavior Evaluation Authority (MEA), that analyses suspicious communication in the VANET, may only need to know whether messages with different pseudonymous identifiers belong to the same vehicle. The task of a MEA is to identify attackers in the network by analyzing misbehavior reports that state non-plausible behavior of vehicles as further detailed in [34], [35] and [53].

In order to fulfill the aforementioned requirements regarding linkability of pseudonyms, we propose a Conditional Pseudonym Resolution Algorithm (CoPRA) that can be integrated into a Public Key Infrastructure (PKI). Using this protocol, pseudonym resolution information can be requested based on defined conditions, i.e. permissions and policies. Depending on the desired resolution information type, several independent authorities are involved in the process in order to avoid misuse. In addition, CoPRA does not decrease the performance and overhead in the vehicular wireless communication as the size of certificates and therefore the message size remains untouched. Our measurements show further that complexity and workload for the pseudonym issuance is not increased. Due to possibly instable communication links and short connection time slots between vehicles and the PKI server, the process of requesting pseudonym certificates can be realized packet-oriented rather than based on complex sessions. Further, we focus in this work on the specific requirements of a VANET (i.e. high speed, delay-sensitive application, high impact in case of misbehavior and strict privacy requirements). Other related network types (e.g. tactical or private MANETs and wireless sensor networks) do not generally have this specific set of requirements.

### 6.2.2 System Model

The ITS model consists of mobile and fixed ITS stations such as vehicles, trucks, roadside stations and PKI servers in the back-end. In order to establish trust between all these entities a Root Certificate Authority (RCA) is established as trusted third party as shown in Fig. 6.2.

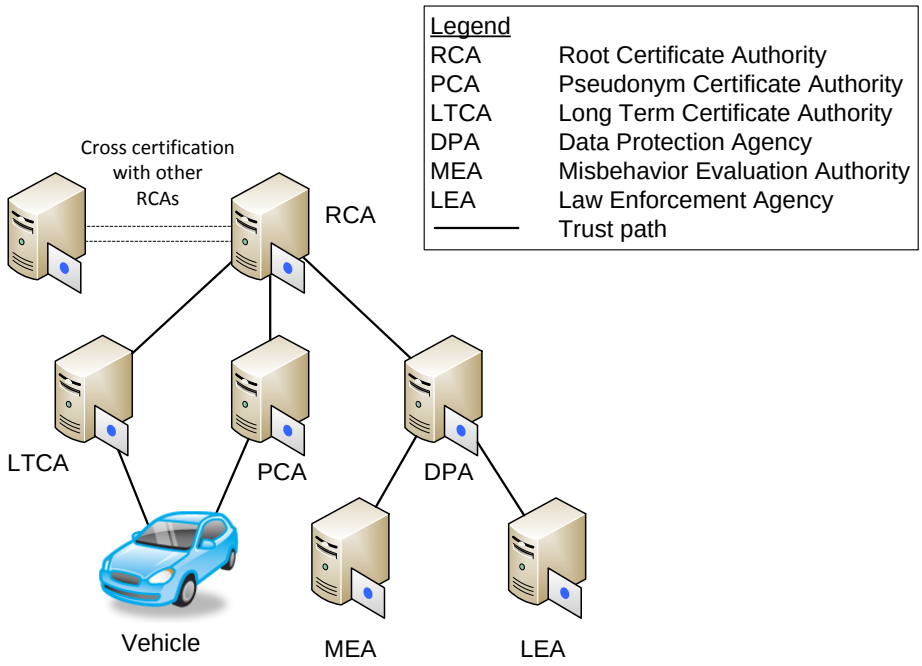



Figure 6.2: Entities of the assumed PKI domain

A Long-Term CA (LTCA) is used to issue a long-term certificate for a vehicle $V$ in the network. In order to protect the driver's privacy, the vehicles are using pseudonymous short-term certificates in the VANET communication that are issued by the Pseudonym CA (PCA). Only a vehicle that is equipped with a valid Long-Term Certificate $LTC$ is able to obtain a short-term Pseudonym Certificate $PC$ from an arbitrary trusted PCA as described in [54] and [55]. After certificate generation, a hashed digest of the related certificate can be calculated to get the long-term identifier $id_{LTC}$ and short-term pseudonym identifier $id_{PC}$ according to [56]. A pseudonym certificate includes a public key $PK_{PC}$ that is related to the private key $SK_{PC}$ but it contains no information linking $id_{PC}$ to $id_{LTC}$. In the phase of certificate issuance, the vehicle needs to communicate with the LTCA and PCA. If a vehicle $V_a$ communicates subsequently with another vehicle $V_b$, it signs outgoing messages with the private key $SK_{PC_a}$ of a short-term pseudonym certificate $PC_a$ and append the related signature as well as the certificate to the message. The receiving vehicle $V_b$ is able to verify the appended certificate $PC_a$ from $V_a$ by checking all authorities up to the RCA in order to trust sent message data. In order to increase the efficiency, the verification of certificates can be omitted if they were previously verified. However,

the message content must be verified every time using the public key $PK_{PC_a}$ from the certificate $PC_a$.

A fundamental information element of the VANET communication is the position of adjacent vehicles. Therefore, a position vector can be found in every beacon message. This vector consists of a short-term pseudonymous identifier $id_{PC}$, an absolute position, a heading value, the current velocity and a related timestamp of sender $V$ which uses the certificate $PC$ at the time interval. Based on the position vectors, every vehicle is running a local misbehavior detection system that verifies the position vector and thereby the neighbor's driving behavior [34,35,57] in order to identify inconsistencies and possible misbehavior. After local evaluation of suspicious behavior the vehicle creates a Misbehavior Report (MR) and sends it to a central Misbehavior Evaluation Authority (MEA) in order to identify the attacker and exclude it [53].

All involved entities of the PKI domain, as shown in Fig. 6.2, are equipped with certificates that are issued by a common trusted root CA. Based on a policy, the RCA puts permissions and authorization information into the certificates that are issued for authorities that would like to resolve pseudonyms for different purposes. A Law Enforcement Agency (LEA) for example may get the permission to request the long-term identifier $id_{LTC}$ whereby a misbehavior evaluation authority gets only the permission to request information whether different pseudonyms belong to the same vehicle. According to Fig. 6.2, a Data Protection Agency (DPA) issues the certificates for the LEA and MEA with appropriate permissions. As long as the PCA and LTCA are not compromised and do not collude in a malicious way, a DPA act as surveillance operator in the pseudonym resolution process.

### 6.2.3 Privacy Preserving Pseudonym Resolution Protocol

The following protocol for pseudonym resolution aims to be applicable in different PKI environments to provide privacy preserving acquisition of pseudonym certificates and enables conditional resolution of pseudonyms in defined situations. Our protocol, named CoPRA, is separated into two processes: During acquisition of pseudonym certificates, resolution information has to be created and distributed as shown in Fig. 6.3 and detailed in Section 6.2.3.1. Subsequently, authorized authorities are allowed to request pseudonym resolution information as described in Section 6.2.3.2. In the resolution process, we further distinguish between a) identity resolution of pseudonyms and b) linkability of pseudonyms.

In case a), an authority $A$ requests the vehicle identity $id_V$ (e.g. license plate number or vehicle identification number) that is related to a given pseudonym $PC$. This identity resolution should be possible only in well defined situations, if for example a law enforcement agency needs to know the identity of a vehicle after a hit-and-run accident. For this purpose, our protocol can be used with a defined number of data protection authorities $DPA_1, ..., DPA_n$ or juridical institutions $J_1, ..., J_n$ that have to be involved in the process to get $id_{LTC}$ and $id_V$. For simplicity, we consider in the following protocol discussions only one instance of a DPA.

In case b), an authority $A$ needs to only get the information whether pseudonyms $PC_{V_{a'}}$ and $PC_{V_{a''}}$ belong to the same vehicle $V_a$. We propose for this linkability resolution a Pseudonymous Long-Term identifier $PLT$ that can be used by a misbehavior evaluation authority to identify vehicles that fake misbehavior events and reports. This kind of resolution may have lower privacy protection requirements, as $id_V$ is not disclosed and $PLT$ can change regularly. Nevertheless, data protection authorities $DPA_1, ..., DPA_n$ can also be integrated in the pseudonym linkability resolution process.

#### 6.2.3.1 Pseudonym Acquisition

The basic protocols for requesting pseudonym certificates from the PKI are described in [54] and follow standardized ETSI specifications [55]. In general, a split of powers between the enrollment authority (LTCA) and the pseudonym certificate provider (PCA) is proposed due to privacy protection requirements inside the PKI. The standard protocols are extended in our proposal in order to make conditional and temporal restricted pseudonym resolution possible. An overview of the protocol is provided in Fig. 6.3 and detailed in Fig. 6.4, whereby the numbers in Fig. 6.3 are related to the steps in Fig. 6.4. The protocol shows the enrollment of vehicles as well as the acquisition of pseudonym certificates.

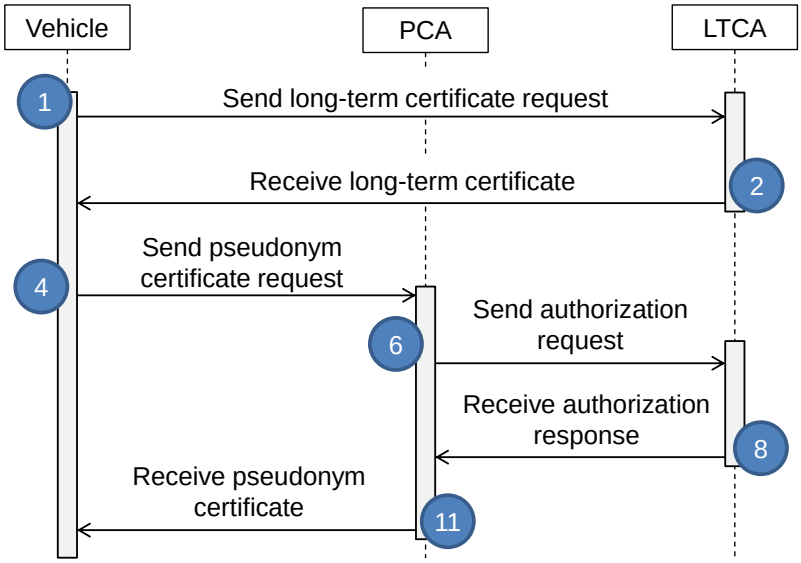



Figure 6.3: Overview of certificate acquisition

We propose a protocol that bases on the well known idea of separation of duties [55,58] in order to protect the identity of vehicles and drivers and ensure unlinkability of pseudonym certificates.

**Enrollment phase** Every vehicle of the VANET has to be equipped with valid certificates in order to communicate with other ITS stations. Therefore, the vehicle $V$ has to

Enrollment phase:

$$V \rightarrow LTCA \quad : \quad (id_V, PK_{LTC_V}) \tag{6.1}$$

$$V \leftarrow LTCA \quad : \quad LTC_V = (PK_{LTC_V}, id_{LTCA}, \sigma_{LTCA}(\circ)) \tag{6.2}$$

Pseudonym acquisition phase:

$$V \quad : \quad req = (PK_{PC_V}, E_{PK_{LTCA}}(id_{LTC_V})) \tag{6.3}$$

$$V \rightarrow PCA \quad : \quad (req, \sigma_{LTC_V}(req)) \tag{6.4}$$

$$PCA \quad : \quad RId_{PC_V} = (\delta(PK_{PC_V})||rand) \tag{6.5}$$

$$PCA \rightarrow LTCA \quad : \quad (\sigma_{LTC_V}(req), \delta(req), RId_{PC_V}, E_{PK_{LTCA}}(id_{LTC_V}), \sigma_{PCA}(\circ)) \tag{6.6}$$

$$LTCA \quad : \quad store(RId_{PC_V}, id_{LTC_V}, id_{PCA}) \tag{6.7}$$

$$PCA \leftarrow LTCA \quad : \quad (\delta(req), exp_{PC_V}, \sigma_{LTCA}(\circ)) \tag{6.8}$$

$$PCA \quad : \quad PC_V = (PK_{PC_V}, id_{PCA}, \sigma_{PCA}(\circ)) \tag{6.9}$$

$$PCA \quad : \quad store(id_{PC_V}, RId_{PC_V}, id_{LTCA}) \tag{6.10}$$

$$V \leftarrow PCA \quad : \quad PC_V \tag{6.11}$$

Figure 6.4: Protocol for issuing long-term and pseudonym certificates

be enrolled at a LTCA in order to get a valid long-term certificate $LTC_V$. Details of the enrollment should be left unspecified in this protocol as vehicle manufacturers may have specific solutions to register their ITS station in a secure manner. Nevertheless, in the first step (1) the enrollment process shall consider authentication, authorization, integrity and non-repudiation of the requesting ITS station in order to prevent enrollment of malicious stations. If this can be assumed the LTCA generates and issues in (2) a new long-term certificate $LTC_V$ based on the given public key $PK_{LTC_V}$. We indicate a signature with the private key $SK_{LTCA}$ over a whole content with $\sigma_{LTCA}(\circ)$. The resulting certificate is sent to $V$ and can be used subsequently to request pseudonym certificates.

**Pseudonym acquisition phase** The protocol for pseudonym certificate acquisition bases on a split of duties between enrollment authority (LTCA) and short-term pseudonym certificate provider (PCA) as proposed in [54]. Vehicle $V$ creates in (3) a pseudonym certificate request that contains the public key of a freshly generated asymmetric key pair $(PK_{PC_V}, SK_{PC_V})$ and the long-term ID $id_{LTC_V}$ that is encrypted with the public key $PK_{LTCA}$ of the LTCA using an Integrated Encryption Scheme (IES). The private key $SK_{PC_V}$ is stored securely in the ITS station and must never leave it. (4) This request is signed with the long-term certificate proving identity $id_{LTC_V}$ and subsequently sent to a PCA. (5) The PCA generates a resolution identifier $RId_{PC_V}$ related to the requested pseu-

donym $PC_V$ by composing the hashed digest $\delta(PK_{PC_V})$ of the given public key $PK_{PC_V}$ and a random $rand$. Inside the PCA domain, $RId$ has to be unique. As the PCA is not able to verify the signature $\sigma_{LTC_V}(req)$ of the pseudonym request, due to the encrypted long-term ID $id_{LTC_V}$, the request is forwarded to the appropriate LTCA. (6) This authentication request consists of the request signature $\sigma_{LTC_V}(req)$ created by $V$, a hash digest of the request $\delta(req)$ created by the PCA, the resolution ID $RId_{PC_V}$, and the encrypted long-term ID $E_{PK_{LTCA}}(id_{LTC_V})$. The PCA signs the authentication request with $SK_{PCA}$ to prove its ownership. We indicate a signature over the whole message with $\sigma(\circ)$. The LTCA decrypts $id_{LTC_V}$ using $SK_{LTCA}$ and verifies $\sigma_{LTC_V}(req)$ with the appropriate public key $PK_{LTC_V}$ to check the correctness of the pseudonym certificate request. Furthermore, the desired pseudonym certificate information like expiration time and permissions are checked by the LTCA. (7) In case of positive verification, the resolution ID $RId_{PC_V}$ is stored in a database of the LTCA linked to the respective long-term ID $id_{LTC_V}$ and PCA identifier $id_{PCA}$. The verification result is further used to generate an appropriate response for the PCA. (8) This response contains, in case of successful verification, a hashed digest of the original pseudonym request $\delta(req)$ and expiration information $exp_{PC_V}$ of the new pseudonym certificate. The whole response message is signed by the LTCA using $SK_{LTCA}$ to prove its possession. (9) After verification of the returned authentication request, the PCA creates a new pseudonym certificate $PC$ and stores the previously generated resolution ID $RId_{PC_V}$ in a database together with the related $id_{PC_V}$ and $id_{LTCA}$ in (10). Finally, the pseudonym certificate $PC_V$ is transmitted to the vehicle in (11).

In order to protect the communication against manipulation and eavesdropping, all data transmitted between the entities in the proposed protocol is encrypted with an IES (e.g. ECIES [59]). Hereby, the sender of a message generates an asymmetric key pair $(PK_{s,r}, SK_{s,r})$ and a symmetric key $K_{s,r}$. This set of keys is only used to protect the message transport between a specific sender $s$ and a receiver $r$ in a session. According to [59], the transmitted message is first encrypted with the symmetric key $K_{s,r}$ and subsequently $K_{s,r}$ is encrypted with the public key of the receiver $PK_r$. This strategy makes atomic communication between the entities (i.e. vehicle, PCA, and LTCA in Fig. 6.4) possible without establishing complex sessions with multiple exchange of packets.

#### 6.2.3.2 Conditional Pseudonym Resolution

Vehicles that are equipped with valid pseudonym certificates are able to use them in VANET communication. In case of misbehavior detection or critical traffic situations (i.e. car accidents) the resolution of the pseudonymous short-term identifier may be necessary. The protocol shown in Fig. 6.5 and detailed in Fig. 6.6 allows linking of different pseudonyms or providing the respective long-term ID of a pseudonym. Based on policies, the LTCA is able to provide different resolution information to an interested authority $A$. A misbehavior evaluation authority $MEA$ may need only temporary linking information of pseudonyms $PC_1, ..., PC_n$ in form of a pseudonymous long-term ID $id_{PLT}$. Whereupon, a law enforcement agency may need to know the non-pseudonymous long-term ID $id_{LTC_V}$ of $PC_V$ in order to request additional information $id_V$ regarding $V$. For our protocol description in Fig. 6.6, we assume the request of the long-term ID $id_{LTC_V}$ by authority $A$ in

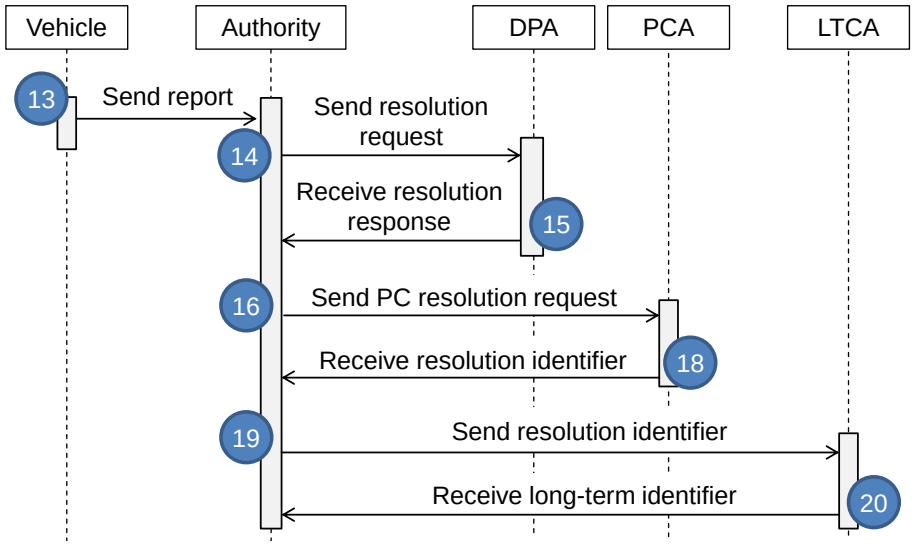



Figure 6.5: Overview of pseudonym certificate resolution

which a DPA must be involved as attesting notary. During communication in the VANET, vehicle $V_a$ or other ITS stations are able to record short-term IDs $id_{PC_V}$ from received messages. (12) According to the motivation of $V_a$, a message $msg$ is created that contains the short-term ID $id_{PC_{V_b}}$ of a vehicle $V_b$ which is involved in an event that triggers generation of $msg$. Additionally, a signed record of vehicle $V_b$ is appended to $msg$ that motivates the pseudonym resolution. This could be for example a broadcasted message containing a position vector proving the existence of $V_b$ at the specific time and position. For simplicity, we add in (12) only one pseudonym that should be resolved. Depending on the purpose, additional short-term IDs with related records can be added to the message $msg$. Before the message is provided to an authorized authority $A$ in (13), the whole message content is signed with the private key of a $PC$ of $V_a$ indicated by $\sigma_{PC_{V_a}}(\circ)$ in our protocol. (14) Based on regulations, defined in a policy, the pseudonym resolution request must optionally be supported by other entities (e.g. data protection agencies DPA). If this support is needed, authority $A$ extracts the pseudonym $PC_{V_b}$ that should be resolved and forwards the original message along with $id_{PC_{V_b}}$ to the respective DPA. Furthermore, the desired resolution type $rt$ (e.g. full identity resolution or pseudonym linking information) is appended. The whole request is signed with the private key $SK_A$ of the authority. Subsequently, the DPA verifies the signature with the public key $PK_A$ and checks whether $A$ is authorized to request pseudonym resolution information from the PKI. (15) If the DPA supports the resolution request, a digest $\delta$ of request data is generated by using a hash function. Subsequently, the digest, the current time $t_c$, and the confirmed resolution type $rt$ are signed and sent to $A$. (16) After receiving the response from the supporting authority, $A$ sends $msg$, $id_{PC_{V_b}}$ and the confirmation from DPA, signed with its private key $SK_A$, to the PCA. (17) The PCA verifies and checks the signatures and permissions of $A$ and $DPA$ and gets the appropriate resolution ID $RId_{PC_{V_b}}$ from its database. In order to prevent misuse of $RId_{PC_{V_b}}$, it is encrypted with the public key of the related LTCA. (18) Subsequently, the PCA generates a response with the digest of message $msg$ and

$$
\begin{aligned}
V_a &: msg = (list(id_{PC_{V_b}}, record_{V_b}, \\
&\quad \sigma_{PC_{V_b}}(record_{V_b})), \sigma_{PC_{V_a}}(\circ)) && (6.12)\\
V_a \rightarrow A &: msg && (6.13)\\
A \rightarrow DPA &: (msg, id_{PC_{V_b}}, rt, \sigma_A(\circ)) && (6.14)\\
A \leftarrow DPA &: res_{DPA} = (\delta(msg, id_{PC_{V_b}}), t_c, \\
&\quad rt, \sigma_{DPA}(\circ)) && (6.15)\\
A \rightarrow PCA &: (msg, id_{PC_{V_b}}, res_{DPA}, rt, \sigma_A(\circ)) && (6.16)\\
PCA &: eRId = E_{PK_{LTCA}}(RId_{PC_{V_b}}, \\
&\quad \delta(msg, id_{PC_{V_b}}), t_e) && (6.17)\\
A \leftarrow PCA &: res_{PCA} = (\delta(msg, id_{PC_{V_b}}), eRId, \\
&\quad rt, res_{DPA}, \sigma_{PCA}(\circ)) && (6.18)\\
A \rightarrow LTCA &: (res_{PCA}, \sigma_A(\circ)) && (6.19)\\
A \leftarrow LTCA &: (\delta(msg, id_{PC_{V_b}}), id_{LTC_{V_b}}, \\
&\quad t_{exp}, \sigma_{LTCA}(\circ)) && (6.20)
\end{aligned}
$$

Figure 6.6: Protocol for conditional pseudonym resolution

the pseudonym ID $id_{PC_{V_b}}$ that should be resolved, the encrypted resolution ID $RId_{PC_{V_b}}$ and the confirmation of DPA. The whole response is signed and sent to $A$. (19) When $A$ receives the data from the PCA, the response $res_{PCA}$ is signed by $A$ and sent to the appropriate LTCA. The ID of the responsible LTCA can be extracted from the encryption header of $eRId$. (20) First, the LTCA verifies all signatures and certificates from $A$, $DPA$ and PCA as well as permissions contained in the respective certificates. Afterwards, the LTCA checks that all contained digests $\delta(msg, id_{PC_{V_b}})$ are equal. The kind of pseudonym resolution is based on the type that must be confirmed by the DPA and the PCA. In the presented protocol we assume a request for full identity resolution. Therefore, the LTCA provides the long-term identifier $id_{LTC_{V_b}}$ that is linked to the given resolution ID $RId_{PC_{V_b}}$. The timestamp $t_{exp}$ denotes the expiry date of the provided long-term identifier. In order to guarantee authenticity and integrity of this information a signature is created by the LTCA over the whole responded data, indicated by $\sigma_{LTCA}(\circ)$.

### 6.2.4 Attacker Model and Security Analysis

In our attacker model, we assume that a single attacker or multiple cooperating attackers that have only access to pseudonymous information (e.g. $PC_V$, $id_{PC_V}$ or $RId_{PC_V}$) aim to get uncontrolled access to the long-term information of a specific vehicle. Alternatively, an attacker aims to get only pseudonym linking information in order to track a specific vehicle within the VANET.

As result, we propose CoPRA that provides a flexible mechanism to conditionally resolve pseudonyms without affecting the privacy of other pseudonyms. Due to the split of duties, one entity alone cannot threaten privacy by linking arbitrarily pseudonyms to the long-term certificate. As PCA and LTCA can verify independently the correctness of requests according to local policies, malicious authorities cannot get arbitrarily resolution information. Only if the following authorities cooperate an unauthorized request would be possible:

- PCA and LTCA are compromised and maliciously cooperate. If both CA types are compromised, a database can be created where both CAs collect linking information between issued pseudonym certificates and related long-term certificates. In this case, the PCA and LTCA are not following the acquisition protocol shown in Fig. 6.4.
- Authority $A$, $DPA$, and PCA are compromised and maliciously cooperate. Assuming the PCA is compromised, arbitrary resolution IDs can be extracted from its database. We propose therefore independent monitoring instances $A$ and $DPA_1, ..., DPA_n$.
- $V_a$, $A$, and $DPA$ are compromised and maliciously cooperate. The report of faked events by $V_a$ is considered in that way, that resolution information is provided based on the event type. Messages $msg$ containing a misbehavior report should only be usable to get pseudonymous long-term IDs and messages $msg$ stating a traffic fatality (e.g. hit-and-run offense) need support by external authorities $DPA_1, ..., DPA_n$ and manual interaction.

The introduction of vulnerabilities to central PKI entities is another aspect that should be analyzed. We discuss resistance of our protocol against important threats: Replay attack, Denial of Service (DoS). The replay of resolution requests sent by external attackers can be detected and directly filtered out at all entities. A digest $\delta(msg, id_{PC_{V_b}})$ is used in this case as unique identifier of a resolution task. It has to be further considered that the $record_{V_b}$, which is part of a message $msg$, contains variable position data and timestamps. Finally, all messages transmitted between the vehicle, authority $A$, $DPA$, PCA and LTCA are signed and encrypted.

The DoS attacks on involved entities can be limited due to the usage of digital signatures. Requests and responses are only accepted and processed if the signature is valid. Therefore, an attacker must spend cryptographic effort in signing operations to mount a DoS attack. Indeed, an attacker could flood the authorities with invalid signed messages. A possible countermeasure is the checking of the sender's certificate first and handle unknown and untrusted senders with lower priority.

### 6.2.5 Application for Misbehavior Detection

For a misbehavior detection and evaluation system it is necessary to get pseudonym linking information in order to identify attackers in ITS communication. A central Misbehavior Evaluation Authority (MEA) collects Misbehavior Reports (MR) from ITS stations of the VANET. As vehicles can change their pseudonyms arbitrarily, it is a major requirement of a MEA to check whether $PCs$ belong to the same ITS station.

The structure of a misbehavior report, shown in Fig. 6.7, contains the type of detected misbehavior, the pseudonymous ID $id_{PC_{V_a}}$ of the reporter node, a list of suspected nodes including their pseudonym IDs $id_{PC_{V_b}}$ and a list of relevant neighbor nodes surrounding the reporter. In every report an evidence of the misbehavior should be added in form of

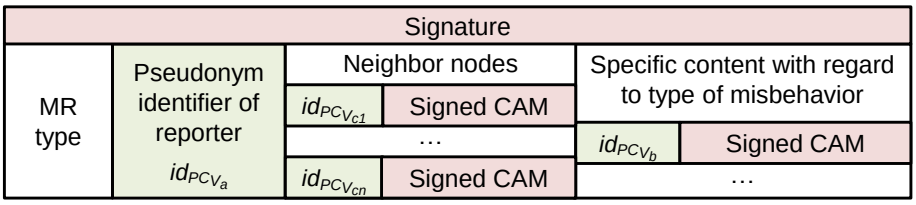



Figure 6.7: Structure of misbehavior report

signed CAMs that attest the existence of the node at the claimed position and time. This signed CAM is used in the protocol by the PCA and possible involved DPAs to verify that a resolution request is justified.

The MEA is further equipped with a certificate that contains permissions to request pseudonym linking information. The certificate of the MEA is issued by a root CA that is trusted by all other involved entities as depicted in Fig. 6.2. Based on the permission contained in the MEA certificate and policies at the PCA and LTCA, a pseudonymous and timely limited identifier $PLT$ is provided by the LTCA. This can be used by the MEA to check if pseudonyms belong to the same sender.

#### 6.2.5.1 Pseudonym Linking for Central Misbehavior Evaluation

The protocol presented in Fig. 6.8 uses specific data for misbehavior evaluation but follows the generic protocol described in Fig. 6.6. In order to balance the system cost, the integration of a DPA is not mandatory for temporal pseudonym linking resolution. However, its integration could be done easily if needed as described in the generic protocol in Section. 6.2.3.2. A vehicle $V_a$ generates in step (21) a MR that contains pseudonymous identifiers of involved ITS stations as depicted in Fig. 6.7 and sends it to the MEA. The received MR is used by the MEA to generate a resolution request in step (22) that is sent to the PCA. We assume in this example that no support of DPAs is required. Based on the MR content, the PCA decides whether the desired resolution type $rt$ is accepted and encrypts the resolution ID (23). The response, that is sent to the MEA in step (24) contains a digest $\delta(MR, id_{PC_{V_b}})$, the encrypted resolution ID and the resolution type. This data is signed by the MEA in (25) and sent to the LTCA. Based on $rt$, the LTCA creates a temporal restricted pseudonymous long-term ID in step (26). This identifier $PLT_{PC_{V_b}}$ is a composed one way hash value containing the long-term ID $id_{LTC_{V_b}}$, a random value $r$ and the expiration time $t_{exp}$. In (27) finally, the digest $\delta$, the resolution ID, and the expiration time of $PLT$ is responded. In order to guarantee authenticity and integrity of this information a signature is created by the LTCA over the whole responded data.

$$
\begin{aligned}
V_a &: MR = (list(id_{PC_{V_b}}, CAM_{V_b}, \\
&\quad \sigma_{PC_{V_b}}(CAM_{V_b})), \sigma_{PC_{V_a}}(\circ)) && (6.21)\\
MEA \rightarrow PCA &: (MR, id_{PC_{V_b}}, rt, \sigma_{MEA}(\circ)) && (6.22)\\
PCA &: eRId = E_{PK_{LTCA}}(RId_{PC_{V_b}}, \\
&: \delta(MR, id_{PC_{V_b}}), t_e) && (6.23)\\
MEA \leftarrow PCA &: res_{PCA} = (\delta(MR, id_{PC_{V_b}}), \\
&\quad eRId, rt, \sigma_{PCA}(\circ)) && (6.24)\\
MEA \rightarrow LTCA &: (res_{PCA}, \sigma_{MEA}(\circ)) && (6.25)\\
LTCA &: PLT_{PC_{V_b}} = (id_{LTC_{V_b}}||r||t_{exp}) && (6.26)\\
MEA \leftarrow LTCA &: (\delta(MR, id_{PC_{V_b}}), PLT_{PC_{V_b}}, t_{exp}, \\
&: \sigma_{LTCA}(\circ)) && (6.27)
\end{aligned}
$$

Figure 6.8: Protocol for temporal restricted pseudonym resolution

#### 6.2.5.2 Comparison of Pseudonym Resolution Protocols

Table 6.1 compares the CoPRA protocol with related schemes for pseudonym resolution in the context of misbehavior detection in ITS communications. In the first row, the effect of pseudonym resolution is compared by means of overhead in pseudonym certificates. As pseudonyms are appended to messages in the wireless communication, the overhead should be as small as possible.

Table 6.1: Comparison of Pseudonym Resolution Schemes for VANETs

| **Topic of comparison** | **V-Token [60]** | **SRAAC [61]** | **CoPRA** |
|---|---|---|---|
| Overhead in pseudonym certificate | $\geq$ 61 Bytes | 0 Bytes | 0 Bytes |
| Certificate acquisition overhead at CA | 0 Bytes | $\geq$ 64 Bytes per cert. | $\geq$ 8 Bytes per cert. |
| Certificate acquisition performance | DSS encryption operation | shared secret interpolations (e.g. [62]) | no additional overhead |
| Certificate acquisition connection type (vehicle $\leftrightarrow$ PCA) | session based (blind signature) [63, 64] | session based (MI-DSS*) [65] | atomic |
| Certificate resolution overhead | $\geq$ 61 Bytes | $\geq$ 64 Bytes | $\geq$ 1 KB |
| Certificate resolution performance | shared secret interpolations (e.g. [62]) | shared secret interpolations (e.g. [62]) | DSS sign and verify operations |

The second row shows the amount of data that needs to be stored at the CAs in order to support pseudonym resolution. In contrast to the V-Token protocol, SRAAC and CoPRA manage the resolution information centrally by storing data in a database. In the third row, the certificate acquisition performance is compared. Here, we consider only operations that are necessary to add resolution information in form of a *V-Token* in [60], a *Tag* in [61] and *Resolution-Id* in CoPRA. In contrast to the related protocols, our scheme entails no cryptographic operations for resolution information generation and storage. The type of connection between vehicle and pseudonym provider is compared in the fourth row. As discussed in Section 6.2.1, the request of pseudonym certificates from the PKI should be packet based. This allows interruption of pseudonym acquisition with later continuation. In row 5 and 6, the overhead and performance in the resolution process is compared. As shown in Table 6.1, our conditional pseudonym resolution protocol does not decrease wireless vehicular communication performance as no additional data is added to pseudonym certificates. Also no additional cryptographic operations are introduced in the pseudonym acquisition phase. We used for evaluations a testbed PKI implementation based on IEEE 1609.2 [56] with LTCA - PCA server separation, running on a quad core CPU with 2.7 GHz. Using this environment, the processing of one pseudonym certificate request takes 179 ms at the CAs and the processing of a request with 50 public keys requires approximately one second. Avoiding additional delay in the pseudonym acquisition phase is important as every vehicle in the network requests regularly hundreds of certificates. The storage of resolution information is in the magnitude of Megabytes and therefore not critical also when several million pseudonym are issued by the PKI. According to row 5 and 6 of Table 6.1, our protocol entails several bytes of data that have to be transmitted between involved entities. Additionally, several signing and verification processes are necessary. But the conditional resolution of pseudonyms is performed relatively seldom compared to the pseudonym acquisition process.

#### 6.2.5.3 Performance Analysis of Pseudonym Resolution

Using the use-case of misbehavior detection, the MEA must check whether identifiers in a misbehavior report belong to separate vehicles. Otherwise, an attacker would be able to send faked misbehavior reports in order to blacklist arbitrary ITS stations. Fig. 6.9 shows the latency in milliseconds of pseudonym resolution processes on the y-axis. On the x-axis the number of pseudonyms to be resolved, contained in a single request, is shown. As discussed in section 6.2.5, a misbehavior report usually contains several pseudonyms $id_{PC}$ from different vehicles (i.e. reporter, suspected nodes, witnesses). In order to prevent misuse and blackmailing, the linkability of involved pseudonyms has to be checked. In Fig. 6.9, the measured latency at involved PKI entities is shown. According to the protocol described in Section 6.2.5.1, the MEA prepares the pseudonym resolution request and sends it to the PCA. Then the PCA checks the content of the request by verifying the contained misbehavior report with included CAMs. This step mainly causes the increase of latency at the PCA with increasing number of desired PC resolutions. The remaining operations at the MEA and LTCA are relatively static. General overhead for every pseudonym resolution is introduced by DSS operations in the protocol. Every message between

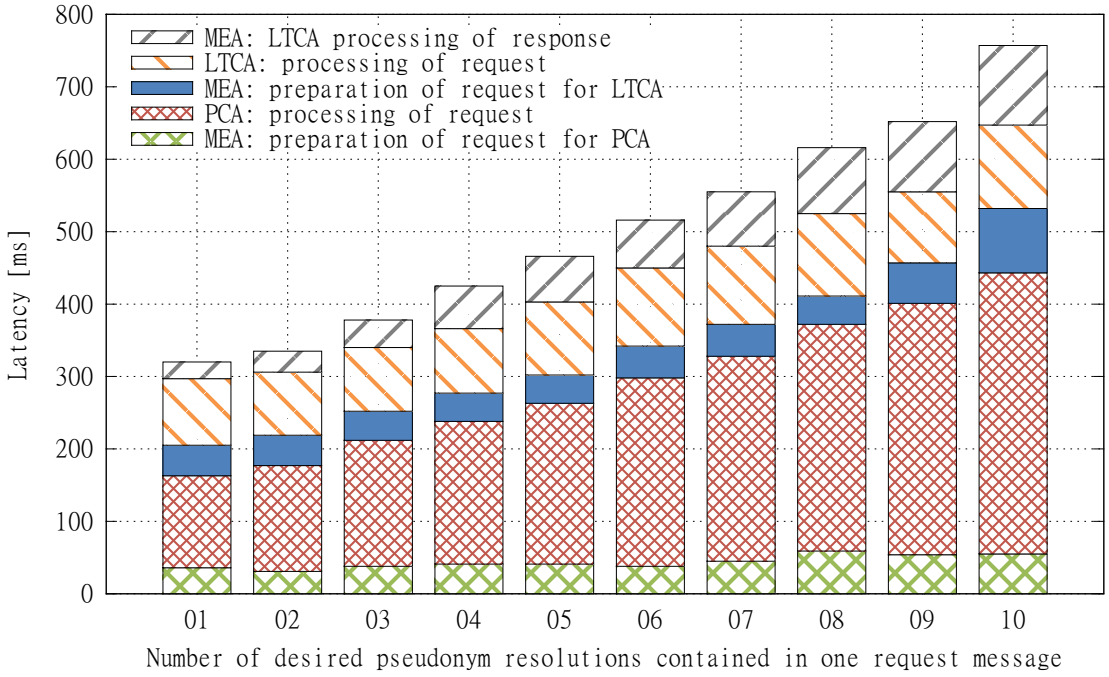



Figure 6.9: Latency distribution in pseudonym resolution with empty database

MEA, PCA and LTCA is signed and encrypted at the sender and decrypted and verified at the receiver using ECDSA and ECIES according to [56].

Fig. 6.10 shows the latency in the pseudonym resolution process with different number of database entries at the MEA, PCA and LTCA. We measured the mean, maximum and minimum latency, as shown on the y-axis, in relation to an increasing number of desired PC resolutions on the x-axis. The more pseudonym certificates are issued by the PCA and LTCA the more database entries are necessary to store the relation between pseudonym ID and resolution ID in the database. As result, the delay for searching the database is increased. But according to Fig. 6.9 and Fig. 6.10, a PKI is able to process approximately 250 pseudonym resolution requests per minute, even if the database is filled. This is sufficient for automated central misbehavior evaluation [53].

### 6.2.6 Conclusion and Outlook

We propose a protocol for conditional pseudonym resolution in VANETs that prevents misuse and preserves privacy and unlinkability of remaining pseudonyms. Focusing on the use-case of misbehavior detection, we have shown that conditional pseudonym resolution is possible without increase of certificate size and therefore increase of bandwidth requirements for wireless communication channels. Our proposed protocol is a balanced solution between full anonymity and uncontrolled arbitrary access to privacy related information (i.e. pseudonym certificate information). The design of our protocol is flexible in order to handle different types of resolution requests motivated by different intentions, for

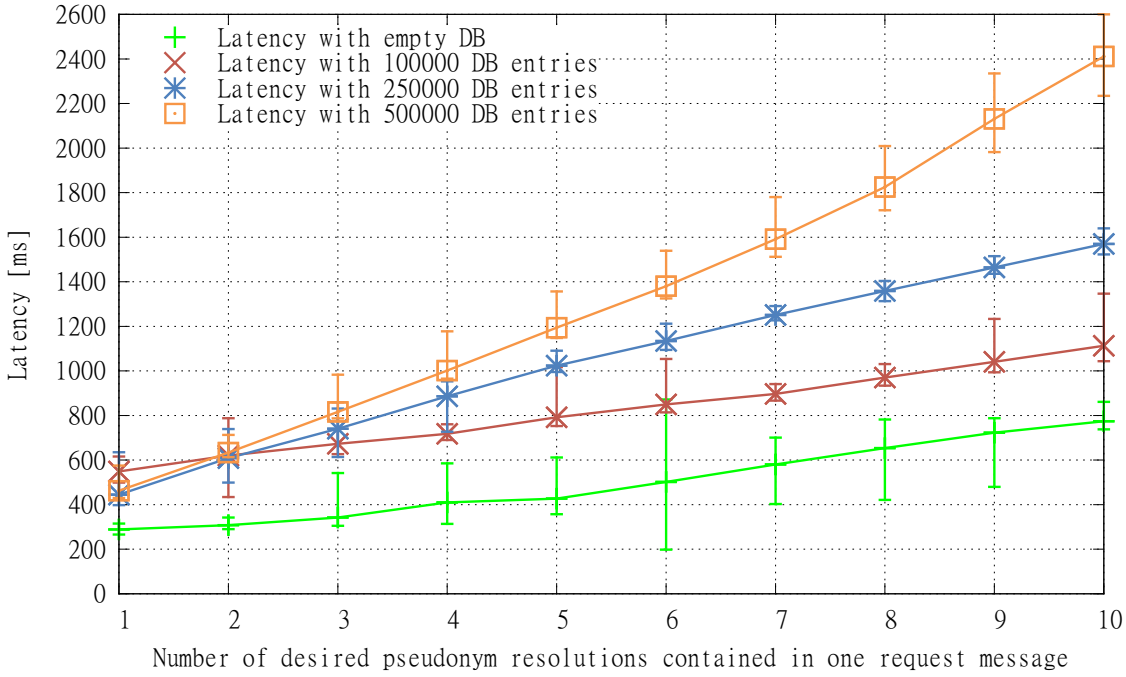



Figure 6.10: Latency of pseudonym resolution related to database size

example lawful interception, misbehavior detection and attacker identification or evaluation of field operational tests. The security analysis has shown the strength of CoPRA as unintended access to pseudonym resolution information is only possible if several CAs, ITS stations and infrastructure agencies cooperate in a malicious way. Our implementation and performance measurements have further shown that CoPRA is not increasing the delay and overhead of pseudonym acquisition and has adequate performance for providing pseudonym resolution information for misbehavior detection and evaluation.

In future work, CoPRA could be extended by trusted computing mechanisms in order to enforce the conformance to the proposed protocols. A policy enforcement scheme could be applied as middleware between CA software and database to restrict and control access to sensitive data.

## 6.3 Vehicular Security and Privacy Architecture

In the paper [66], we present the first implementation of a Vehicular PKI (VPKI), in order to secure V2X using a privacy-preserving architecture according to the standards. We present a *kerberized* version of a VPKI using cryptographic tickets to enable Authentication, Authorization and Accountability (AAA) to the provided services. Our scheme offers credential management, while preserving the privacy against the VPKI itself. Finally, we present an efficiency evaluation of our implementation and demonstrate its applicability.

### 6.3.1 Problem Statement

Each vehicle is equipped with a tamper-resistant crypto-module able to perform advanced cryptographic operations, such as to digitally sign and encrypt messages. All packets transmitted by the vehicles should be authenticated. Packet authentication is not a guarantee of correctness, but the hardware security module greatly improves security as it reduces the chances of cryptographic keys being stolen. Each vehicle frequently broadcasts safety messages.

We consider adversaries that deviate from the expected operation of the V2X protocols and can harm the security of the system and the privacy of its users in various ways. Launching impersonation attacks, the attacker claims to possess a legitimate identity and can fabricate messages or replay old packets. Attackers can deliberately change the content of packets to achieve erroneous or malicious behaviour. Such packet forgery attacks can result in serious implications for V2X especially when targeting safety beacons. Moreover, adversaries might try to gain access to V2X services, and eventually obtain valid credentials, for example pseudonyms. Non-repudiation is an important security property for V2X, especially for accountability purposes. Jamming in V2X is a *low effort* attack that can be launched over small or wider geographical areas, but is out of the scope for the paper. Adversaries targeting vehicle privacy and anonymity by linking successive pseudonyms, can leverage on the information included in safety-beacons, in order to reconstruct real vehicles' whereabouts. For this, academia, industry, and standardization bodies have converged on the use of pseudonymous credentials for privacy protection. Moreover, privacy needs to be considered even in the presence of untrusted (i.e. honest but curious) infrastructure and misbehaving users. In the later case, the anonymity provided by the pseudonymous identifiers needs to be revoked.

All of the above underline the importance of secure and privacy-preserving credential management for safety applications in V2X. Nevertheless, given the near-deployment status of V2X, a whole ecosystem of non-safety services and applications is on the way. To facilitate their adoption by users, a VPKI must offer them security (i.e. AAA services) and protect the privacy of travellers/users against inference attacks and profiling. All these define the need for a scalable, modular and resilient VPKI implementation whose services support, but can be extended beyond, the domain of safety-applications. This becomes critical given the absence of an implementation and evaluation of such an infrastructure. These points comprise the motivation and the scope of our work. We design, implement and evaluate a standard-compliant VPKI, able to accommodate the security and privacy requirements for safety applications and to offer secure and privacy-preserving credential management to any other vehicular application.

### 6.3.2 The VPKI Architecture

In this section we present our architecture and the relevant protocols. We focus on the security and privacy aspects of our approach, and define a privacy-preserving pseudonym acquisition protocol which can be easily extended to support other vehicular services.

#### 6.3.2.1 Security & Privacy Discussion

Packets signed under the private key of the vehicle, residing inside the hardware security module, are then transmitted along with the corresponding certificate. The VPKI architecture should support key management and certificate distribution, thus ensuring (i) V2X message integrity, (ii) message & vehicle authentication in both V2I and V2V, and (iii) non-repudiation of origin security properties. Vehicles can establish secure channels (e.g., using TLS tunnels), thus achieving confidentiality against external eavesdroppers. Authorization and accountability is accomplished using *tickets*; that is reusable proofs of access rights to a given service. Tickets are signed by a trusted authority to avoid forgery and integrity attacks as presented in Sec. 6.3.2.3. We now discuss the usefulness of two types of certificates:

**Pseudonyms.** In order to preserve location privacy and anonymity in V2X, each vehicle possesses a set of short-lived pseudonyms, obtained by a trusted pseudonym provider. Each pseudonym has a lifetime ranging from seconds to hours, defined by the pseudonyms provider. A vehicle can decide to change the active pseudonym in order to prevent the tracking of its location. Safety beacons are digitally signed under the current pseudonym identity. By increasing the frequency of pseudonym changes, the chances for an adversary to launch a successful attack against privacy are reduced.

**Long-term Certificates.** A pseudonym acquisition protocol is necessary to obtain new sets of pseudonyms when the old ones are close to expire or have been already used. However for accountability and authorization purposes, the vehicle needs to be authenticated using its long-term identifier and then obtain anonymous authorization credentials, in the form of tickets. For this reason, each vehicle should be able to prove its real identity using a *long-term identity*.

#### 6.3.2.2 Architecture Proposal

Our scheme comprises the following three trusted CAs, according to the terminology used in [67] and compatible with the definitions in [68]:

- **LTCA**:
  The LTCA is the issuer of the vehicle's long-term certificates and tickets.
- **PCA**:
  The PCA is the provider of the vehicle's pseudonyms.
- **Resolution Authority (RA)**:
  The RA *de-anonymizes* pseudonymous certificates in case of misbehaviour detection.

The long-term certificate is a digital signature of the LTCA over a set of vehicle-specific identifying data, a validity period $[t_s, t_e]$, and the vehicle's long-term public key $K_v$:

$$LT_v = Sig_{\mathsf{LTCA}}(K_v, \mathsf{data}_v, [t_s, t_e])$$

We assume that each vehicle $v$ has a long-term certificate $LT_v$ and the corresponding private key $k_v$ pre-installed in its hardware security module, as proposed in [69]. The vehicle also obtains and stores a set of pseudonyms of the following form:

$$P_v^i = Sig_{\text{PCA}}(K_v^i, [t_s, t_e])$$

Pseudonyms also have a specified validity period $[t_s, t_e]$ and contain a public key $K_v^i$ for verification.

#### 6.3.2.3 Pseudonym Request Protocol

We now describe the protocol for the vehicles to obtain pseudonyms from the PCA. All communications are performed over a secure TLS tunnel, which guarantees confidentiality against external adversaries, and prevents tickets hijacking. For vehicle-to-PCA communications one-way authentication of the server to the vehicle is used, in order to preserve the anonymity of the vehicle. In a nutshell, the protocol starts with the vehicle being authenticated by the LTCA using its long-term credentials to obtain a *ticket*. The ticket, $tkt$, does not contain any data attributable to the vehicle and it is of the form:

$$tkt = Sig_{\text{LTCA}}([t_s, t_e], \{S_1\}, \ldots, \{S_n\}),$$

where $[t_s, t_e]$ is the ticket validity period and $S_i$ is a generic service identifier. By ensuring that $t_e$ does not exceed the subscription expiration time for any of the $S_i$ included in $tkt$, the LTCA can guarantee that service subscription periods are not violated.

$$V \longrightarrow LTCA : Sig_{k_v}(t_1, \text{Request}) \parallel LT_v \tag{6.28a}$$
$$LTCA \longrightarrow V : tkt \tag{6.28b}$$

Initially, the vehicle issues a ticket request to the LTCA in order to obtain access to the PCA. The LTCA checks the validity of the request, generates $tkt$ and sends it back to the vehicle. The vehicle then generates a set of private/public key pairs $(k_v^i, K_v^i)$ inside its hardware security module and sends the public keys $K_v^i$, along with $tkt$, to the PCA.

$$V \longrightarrow PCA : t_3, tkt, \{K_v^1, \ldots, K_v^n\} \tag{6.29a}$$
$$PCA \longrightarrow V : t_4, \{P_v^1, \ldots, P_v^n\} \tag{6.29b}$$

The PCA assesses the validity of the ticket and signs the received public keys $K_v^i$ using its private key. The pseudonyms $P_v^i$ are then sent back to the vehicle. The same ticket can be re-used for multiple pseudonym requests, or different service providers during its validity period.

**Unlinkability of requests.** We avoid signing pseudonym requests under the long-term or the current-pseudonym identities of the vehicle. In both cases the PCA can breach vehicle privacy. In the first case, linking the issued pseudonyms to the long-term identifier is trivial; in the latter case, the PCA is able to link the new set of issued pseudonyms with the one used for the request. Therefore the PCA can link sets of pseudonyms and thus, compromise privacy. On the other hand, using a new *ticket-per-request* can effectively

protect vehicle privacy against the PCA, since no linking is possible between the ticket, the long-term certificate, or any of the pseudonyms. Moreover, the vehicle can issue a request per pseudonym, thus restricting the ability of PCA to link pseudonyms within a request. The proof of the unlinkability is straightforward and omitted here due to space limitations.

#### 6.3.2.4 Pseudonym & Token Revocation

Pseudonyms and long-term certificates should be revoked in a number of different scenarios: for example when a vehicle is involved in an accident or misbehaves. Similarly, a ticket can be revoked to deny access to the service e.g., in case the ticket should not be reused. In order to keep the network *up-to-date* in terms of the status of revoked certificates and tickets, Certificate Revocation Lists (CRLs) are used. Revocation lists are publicly available, so that every entity in the V2X network has access to them. CRLs are digitally signed with the private key of the authority that issues them. The PCA signs the revocation lists containing the revoked pseudonyms and the LTCA the CRLs containing the long-term certificates. The dissemination of the CRLs is orthogonal to the work presented in [66]. Equivalently, TRL can be used for ticket revocation, published by the LTCA in case of ticket revocation. We omit further discussions on ticket and certificate revocation in the work because of the limited space.

#### 6.3.2.5 Resolution Protocol

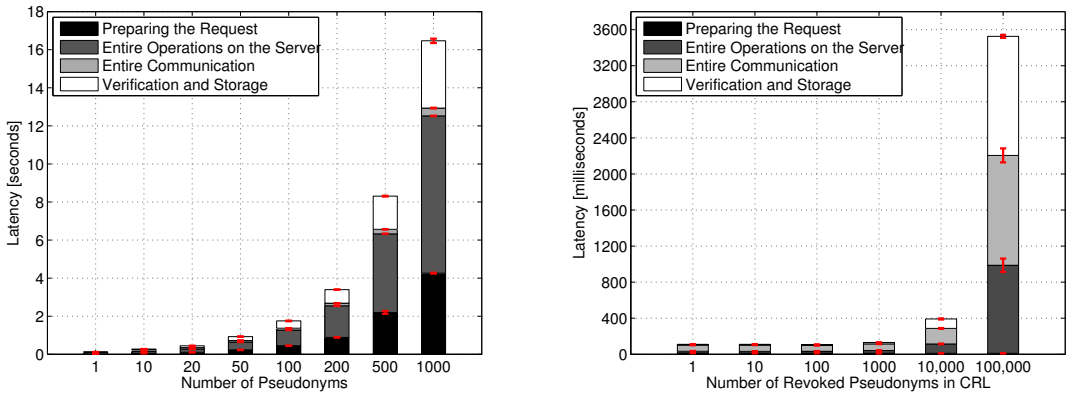
Due to the safety critical nature of V2X, the revocation of anonymous credentials is not sufficient *per se* and complete vehicle de-anonymization is required. The resolution protocol is executed with the RA acting as a coordinator between the PCA and the LTCA. The PCA reveals to the RA the link between the pseudonyms and the anonymous ticket. Then, the LTCA reveals the link between the the ticket the vehicle's real identity. Therefore, the RA can combine both pieces of information and perform the resolution.

The RA generates a digitally signed *resolution request* to the PCA. The request includes the pseudonym $P_v^i$ (or the set of pseudonyms) that have to be resolved. The PCA retrieves all the pseudonyms that were issued as a result of the same vehicle pseudonym acquisition request from its database, along with the corresponding ticket $tkt$.

$$RA \longrightarrow PCA : Sig_{\mathsf{RA}}(P_v^i, t_1) \tag{6.30a}$$
$$PCA \longrightarrow RA : Sig_{\mathsf{PCA}}(tkt, t_2) \tag{6.30b}$$

Having received the ticket $tkt$ the RA forwards it to the LTCA, which can in turn reveal the corresponding long-term identity of the vehicle. Mappings between issued tickets and the corresponding long-term identifiers exist in the database of the LTCA.

$$RA \longrightarrow LTCA : Sig_{\mathsf{RA}}(tkt, t_3) \tag{6.30c}$$
$$LTCA \longrightarrow RA : Sig_{\mathsf{LTCA}}(LT_v, t_4) \tag{6.30d}$$




(a) Latency to obtain pseudonyms in seconds (per vehicle).

(b) Latency to obtain CRLs (per vehicle).

Figure 6.11: Performance evaluation of the VeSPA protocol.

With the completion of the protocol, the long term identity $LT_v$ is resolved and the vehicle's pseudonyms have been revoked. Revocation is performed according to the previous section, which will eventually evict the vehicle from the V2X network. The LTCA should also invalidate the received tickets by including them in the Ticket Revocation List, to prevent adversaries from distributing tickets among each-other.

### 6.3.3 Results

In this section we present the performance of the proposed VPKI architecture. CAs were implemented using OpenCA, on separate servers equipped with an Intel Xeon Dual-Core $3.4$ GHz processor and $8$ Gbytes of RAM. All V2I and Infrastructure to Infrastructure links are secured with TLS, while the study of the communication channels are out of the scope of the paper. ECC-256 keys are used for both infrastructure and vehicle certificates. Our implementation is compatible with the IEEE 1609.2 draft proposal [56]. The ticket size is $498$ bytes and the pseudonym size is $2.1$ KBytes.

**Vehicle: Pseudonym Request.** In Fig. 6.11a, we present latency results for acquiring a set of of pseudonyms from the PCA. The vehicle needs $73,4$ ms to obtain a new ticket from the LTCA (eq. 6.28). To acquire one pseudonym the vehicle needs $120$ ms and $3\,400$ ms for $200$ pseudonyms (eq. 6.29). For requests of $1\,000$ pseudonyms, which should be sufficient for a relatively long period or time (e.g., for a day if the pseudonym lifetime is around 1 minute), we observe that the total latency is $16\,460$ ms. $50\%$ of the total latency concerns PCA side operations, and $26\%$ is devoted on the preparation of the query, for examples the creation of private/public keys and digital signatures over the public keys. The preparation of the request can take place off-line, which can eventually reduce the total time by $4\,260$ ms (darkest bar in Fig. 6.11a). Excluding the verification and storage time that occurs at the vehicle, the total processing time (communication plus operation on the server) to obtain $1\,000$ pseudonyms is reduced to $8\,670$ ms. Results suggest that

our approach is efficient. Additionally, taking into consideration the fact that the vehicles will be equipped with hardware accelerators [67], we can conclude that the time required for a vehicle to obtain a pseudonym will be significantly reduced.

| Pseudonyms Req. | 1 | 100 | 1.000 | 5.000 | 20.000 |
|---|---|---|---|---|---|
| Signature Ver. | $0,004$ | $0,361$ | $3,3618$ | $18,09$ | $72,33$ |
| Pseudonyms Gen. | $0,004$ | $0,349$ | $3,34$ | $17,72$ | $70,9$ |
| Total Time | $0,02$ | $0,817$ | $8,826$ | $41,672$ | $167,3$ |

Table 6.2: Latency to issue pseudonyms in seconds by the PCA

**PCA: Pseudonym Issuance.** Table 6.2 shows the time needed by the PCA to process pseudonym requests from vehicles. The processing time includes the verification of the received request (including ticket verification), pseudonym generation time and other relevant PCA operations (e.g., storage and handling of the received public keys). For a total of $5\,000$ pseudonym requests issued by multiple vehicles, $41\,672$ ms are needed. For $20\,000$ pseudonyms the server needs $167,300$ ms. It is straightforward that the pseudonym's lifetime is a determinant factor for the PCA's workload.

**CRL Distribution.** Fig. 6.11b shows the time needed by a vehicle to obtain the CRLs of revoked pseudonyms. The preparation of the request by a vehicle takes $11$ msecs. The *Server Operations* time corresponds to the generation of the CRL (including signing it) at the PCA. We observe that latency increases with the number of entries in the CRL. For large chunks of information (e.g., $100\,000$ entries in the CRL) the communication time is an important fraction of the total time; $1\,218$ ms for $100\,000$ entries in the CRL. For the latter case, the verification of the PCA's signature and the storage of the obtained CRL, can take up to $1\,324$ ms.

| Pseudonyms Resolved | 1 | 10 | 50 | 100 | 200 |
|---|---|---|---|---|---|
| Pseudonyms Prov. (PCA) | 73 | 135 | 304 | 516 | 922 |
| Identity Prov. (LTCA) | 9 | 10 | 15 | 20 | 57 |
| Resolution Auth. (RA) | 265 | 348 | 604 | 916 | 1598 |

Table 6.3: Resolution latencies in milliseconds; PCA, LTCA & RA

**Certificate Resolution.** Certificate resolution (eq. 6.30) times are presented in Table 6.3. Calculation times include server side operations (e.g., fetching the requested certificate from the database), sign and publish the certification list. The LTCA has the lowest overhead, since the number of tickets is less than the number of pseudonyms that need to be retrieved from the databases of the LTCA and PCA respectively. The resolution of $200$ pseudonyms takes less than $1\,000$ ms for the the PCA, and we believe that our resolution protocol does not introduce a significant overhead for the VPKI. The RA has the highest workload during the resolution process ranging from $265$ ms (for $1$ pseudonym) to $1\,598$ ms (for $200$ pseudonyms).

### 6.3.4 Conclusion

In the paper [66] we presented the implementation of a distributed VPKI architecture, in order to provide security and privacy protection in V2X. We proposed the use of tickets to guarantee unlinkability between consecutive vehicle requests for pseudonyms, when a

new ticket is used for each request. To the best of our knowledge, that is the first work that provides AAA capabilities for a VPKI according to the current standards and the privacy requirements. Part of our future work includes the integration of relevant privacy-preserving methods and anonymous authentication techniques in our protocols. We believe that our scheme is efficient, applicable and thus, it can pave the road towards secure and privacy preserving V2X.

## 6.4 Towards a Secure and Privacy-preserving Multi-service Architecture

The need to grant fine-grained access, across multiple domains, to the a multiplicity of diverse services increases complexity dramatically, making it hard to address with the current identity and credential management facilities alone

Our proposal [70], which we term a *multi-service* security and privacy-enhancing architecture for V2X, seeks to address this challenge. We leverage long-term credential and identity managing entities, expected to be deployed for V2X. We extend their mandate to handle the authorization of registered vehicles for specific services. To enable access, we leverage another longer-standing concept, a *ticket*, and cater to multi- and cross-domain operation. With these design choices, while being standard-compliant [56, 68], our architecture allows efficient and fine-grained access control in a privacy-enhancing manner. At the same time, it greatly simplifies the tasks of the service providers, and it can be further extended by leveraging web services; as a result, it can facilitate deployment of services and contribute to the enrichment of V2X functionality.

### 6.4.1 Problem Statement

A *large-scale* deployment of V2X systems is expected, with numerous LTCAs, PCAs, Root CAs, and Authentication Authorities (AAs). This deployment can be pretty *diverse*; these entities could be instantiated by state authorities, local governments, counties, cantons, metropolitan areas, cities, constituting a forest of hierarchies. At the same time, car manufacturers or any other private party (e.g., the same way that certification authorities are run in the traditional wire-line Internet) could instantiate them. For simplicity, let us term a subset of such entities and the registered with them vehicles as a V2X system *domain*.

At the same time, scores of new services are expected, along with increased connectivity of the vehicle to the (rest of) the Internet. The diversity of these services will be much higher than that of the V2X system security entities: potentially anyone could offer any service to an Internet-enabled vehicle, equipped with multiple radios. Of course, the main stake-holders (car manufacturers, transportation authorities, cities, telecommunication providers) are expected to provide a plethora of V2X-specific services. Similar services could be addressed to users within a specific or across domains; each vehicle could access any set of services; Service Providers (SPs) active in different domains

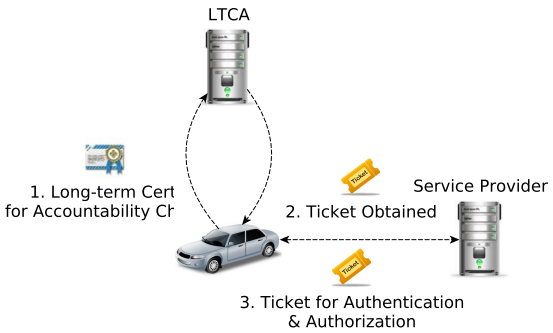



Figure 6.12: VeSPA: Granting access to a service

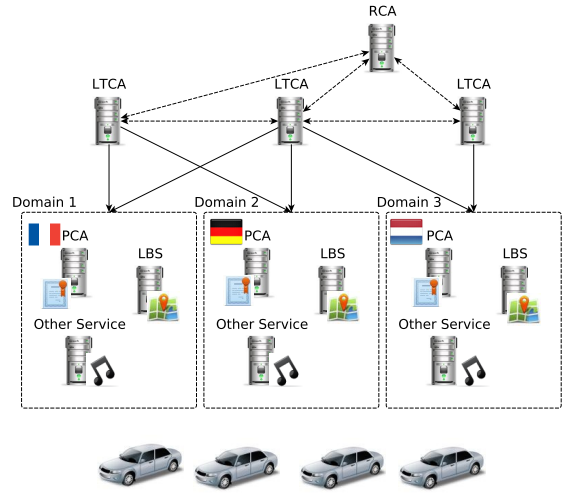



Figure 6.13: Multi-Domain & Multi-Service Architecture

could have service agreements for their users. The question rises naturally: *In this V2X landscape, how can a vehicle access efficiently and effectively any service it is entitled to, within any domain?*

A straightforward answer can be that each SP authenticates each vehicle and grants access. This would incur high complexity for the SPs, while identity and credential management facilities are already planned for V2X. It would then be natural to leverage these facilities: a vehicle could be authenticated and granted access based on its long-term keys and credentials. Nonetheless, this would imply loss of privacy, as all accesses would be linkable. The alternative would be to use short-term keys and credentials. This would be accountable yet only allow coarse-grained access control: for example, any pseudonym from a PCA provides access to a said service. But this would go against the provision of differentiated services to users.

What we are after: (i) fine-grained access control, (ii) privacy-preserving and (iii) accountable service access, (iv) flexible, interoperable, scalable multi-domain operation, (v) reuse of existing Vehicular PKIs (VPKIs) and the achieved protection, and (vi) standard compliance. Moreover, we want a solution that does not add complexity on the SPs, to facilitate deployment of the foreseen multiplicity of services.

### 6.4.2 VeSPA: A Kerberized VPKI

To address the requirements outlined in Sec. 6.4.1 and move towards a *multi-service* architecture for secure V2X, we make the following basic design choices. We de-couple the system entity responsible for access control decisions, the Policy Decision Point (PDP), from the Policy Enforcement Point (PEP) [71], the entity that enforces policy decisions. In the context of a VPKI, the PDP is the LTCA and the PEP is the PCA [1]. Then, we

use the long-known concept of a *ticket* as an enabler of access, inspired by the *Kerberos* protocol [72].

We extend our Vehicular Security and Privacy-preserving Architecture (VeSPA), a VPKI architecture that uses *tickets* for Authentication Authorization and Access Control and combines V2X standards [56] and current prototypes [67] into a unified design. Extending [66], we present how VeSPA can treat multiple services and support them across different domains. VeSPA handles vehicles as clients who hold authorization tickets, in a similar manner to Kerberos. VeSPA achieves (i) *Authentication* of each vehicle to the infrastructure, (ii) *Authorization* of the vehicle to access the offered services, and (iii) *Accountability* of the vehicle for the accessed services, using the tickets and the long term credentials of the vehicles. Finally, VeSPA achieves enhanced privacy protection against the infrastructure by making any two service requests of the same vehicle unlinkable by the SP (e.g., the PCA). For the description of the protocols that follow, we assume that all communications take place over a secure TLS channel.

#### 6.4.2.1 Obtaining Tickets

To access a service, vehicles have to obtain a valid ticket first. The vehicle establishes a secure communication channel with the LTCA, which acts as the authentication and authorization point of the VPKI and, therefore, the issuer of the tickets. Vehicles are authenticated using their long-term certificates in order to provide accountability for the services. Each *ticket request* includes the list of services the vehicle wishes to access. The LTCA is responsible for verifying the ticket request, by checking whether the vehicle should be given access to services included in the request. Reasons to reject a ticket request include an unpaid subscription to services, an invalid vehicle digital signature, or an already issued ticket for a requested service.

Tickets are digitally signed by the LTCA. The lifetime of a ticket is defined by the LTCA in the ticket itself. The ticket format is:

$$tkt = Sig_{\text{LTCA}}(t_e, \{S_1\}, \ldots, \{S_n\}),$$

where $t_e$ is the ticket's expiration time and $S_i$ is a generic service identifier. By ensuring that $t_e$ does not exceed the subscription expiration time for any of the $S_i$ in $tkt$, the LTCA can guarantee that service subscription periods are not violated. A ticket request can be made for each of the services that the vehicle subscribes to, or alternatively for a set of those, depending on the preferred level of anonymity. Separate ticket per service can enhance privacy, as Service Providers cannot learn user profiles.

Protocol 6.28 allows the vehicle to obtain tickets from the LTCA:

$$V \longrightarrow LTCA : Sig_{k_v}(t_1, S_1 ... S_i) \parallel LT_v \tag{6.31a}$$

$$LTCA \longrightarrow V : tkt \tag{6.31b}$$

#### 6.4.2.2 Accessing the Service

Having obtained the ticket, the vehicle holds a (reusable) proof of access rights to a list of services (in the ticket signed by the LTCA). For example, consider a vehicle requesting access to a Location-Based Service (LBS). The ticket request contains the identity of the LBS (Steps 6.31a and 6.31b). The LTCA verifies the requesting vehicle does not already hold a valid ticket for the specific LBS, to avoid sybil attacks against service providers. If vehicles obtained and hold tickets from earlier executions of Protocol 6.31, they can directly get authorized to the LBS and skip the ticket obtaining phase.

Eventually, the ticket will be presented to the LBS provider by the vehicle, both as a proof of a successful authentication and authorization to the infrastructure. The LBS server checks the validity of the ticket by verifying the LTCA's signature, the ticket's lifetime, and if the LBS service is listed in the ticket. The overview of an access request to a vehicular service is given in Fig. 6.12. Communication with the SP is done over a TLS tunnel, using one-way, server to vehicle authentication.

#### 6.4.2.3 Multi-Domain Architecture

A VPKI is expected to cover a domain, thus an LTCA should support thousands of registered vehicles. However, vehicles cannot be geographically restricted and services should be supported across multiple domains. Fig. 6.13 shows a case of different VPKI domains in three countries, with multiple services offered within each domain. A French car registered to a French VPKI, may travel to an area corresponding to the German domain, but still request access to services offered on a global scale; for example a LBS service that delivers real-time data to the vehicles. Even if the same service is offered across multiple domains, it might be subject to different conditions in each domain, e.g., an increased commission for services outside the native (home) domain or different policies altogether.

VeSPA can support vehicular applications in multiple domains using the tickets as anonymous proofs of access rights across federations of VPKIs. As shown in Protocol 6.32, the vehicle first obtains a native ticket from $LTCA_A$ in its *native* VPKI domain. By leveraging on the trust association between $LTCA_A$ and $LTCA_B$, the native ticket can then be exchanged for a new one, obtained from the *foreign* domain's $LTCA_B$, in a similar approach to multi-realm Kerberos.

Continuing the previous example, the French car should first obtain a valid ticket from the French domain, if it doesn't already have a valid one. The German domain can then verify the validity of the ticket presented by the French car (e.g., if the requested service is offered in its own domain) and eventually apply its own policies regarding the requested service. Finally, a new ticket is issued by the German domain and sent to the French car, which can now continue accessing the service in the German domain. This way, VeSPA can support vehicular applications with multi-domain Authentication Authorization and Access Control, while employing *domain-specific* policies for each service (by including those in the ticket).

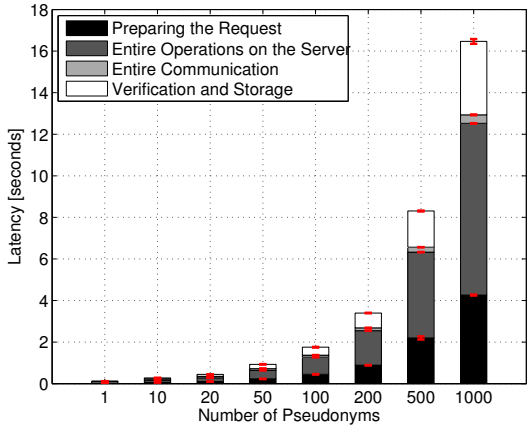



Figure 6.14: VeSPA: Latency in obtaining pseudonyms

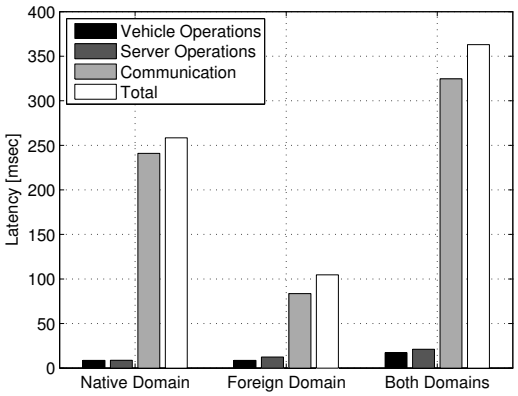



Figure 6.15: Performance of the Multi-Domain AAA Protocol

$$V \longrightarrow LTCA_A : Sig_{k_v}(t_1, S_1...S_i, Dom_B) \parallel LT_v \tag{6.32a}$$
$$LTCA_A \longrightarrow V : tkt_A \tag{6.32b}$$
$$V \longrightarrow LTCA_B : t_2, tkt \tag{6.32c}$$
$$LTCA_B \longrightarrow V : tkt_B \tag{6.32d}$$

### 6.4.3 Efficiency Analysis

We use the same experimental setup as in [66]. The average time for a vehicle to obtain one ticket containing a single service identifier from the LTCA is $100.95$ msec. This low latency indicates that VeSPA can efficiently facilitate operations with *one ticket per service*, an approach for enhanced privacy protection. Moreover, VeSPA supports all the currently proposed VPKI protocols for certificate management, such as pseudonym acquisition and CRL distribution.

The pseudonym acquisition protocol incurs significant overhead; the higher the sought unlinkability, the higher the number of pseudonyms needed. Fig. 6.14, shows the latency for each vehicle to obtain a certain amount of pseudonyms. Acquisition of $1000$ pseudonyms has an average latency of $16.46$ sec. The number of requested pseudonyms depends on desired location privacy and the PCA policy. Nevertheless, $1000$ pseudonyms are considered sufficient by the VC community for a period of one day, either by using equal validity time per pseudonym, or shorter pseudonym lifetime for *high mobility* hours and longer pseudonym lifetime for *low mobility* hours.

The pseudonym resolution adds very low latency to the VPKI operation: for the resolution of $200$ pseudonyms, the PCA and the LTCA need $922$ msec and $55$ msec respectively. Furthermore, the Multi-Domain operation protocol also incurs low latency. The vehicle

has to establish a secure connection with its distant native LTCA server while in a foreign domain; this communication has the dominant latency. For our experiments, we measured the latency to establish a TLS connection with a server that is $1300$ km away. The total communication costs are $258.4$ msec for the native domain and $104$ msec for the foreign domain. The foreign LTCA has the additional overhead of verifying and handling the ticket presented by the vehicle, compared to the computational cost for the native LTCA, which only has to issue the ticket. Overall, the multi-domain protocol has a latency of $363$ msec, which shows its efficiency and applicability for future V2X systems.

### 6.4.4 Future Directions for VPKIs

There are alternative ways of performing identity management, leveraging well-defined open standards and solutions currently in use for traditional networks. More specifically, we are developing an instantiation of an architecture structured around the Web-Services paradigm, where the LTCA serves as the Identity Provider (IdP) and the PCA as a SP. This way we treat the provision of pseudonyms, and consequently privacy, as a service.

#### 6.4.4.1 Identity Management in a Web Services-based VPKI

In a Service-oriented-Approach, an IdP is responsible for operations such as user registration, issuance of long-term certificates, user revocation and enforcement of security policies (i.e., authorization and access control). By following a Web Services (WS) approach, in the context of V2X, the LTCA becomes an IdP, and as a result, all of the aforementioned services can be transparently offered to any SP, including the PCA.

The merging of WS with V2X can yield numerous benefits, especially in the context of trust-establishment. More specifically, to establish trust between the IdP and the SPs a WS-Metadata exchange needs to take place. In principle, metadata are XML based entity descriptors. They contain various pieces of information, such as authentication requirements, the URI of the VPKI entities, protocol bindings and most importantly, digital certificates. For example, referring to Figure 6.13, each SP may opt to establish trust relations with multiple IdPs. An SP can exchange metadata with multiple IdPs and vice versa. This approach could allow the construction of a complex Web of Trust in a manner that satisfies policies and trust relationships without the need for Root CA. Unlike traditional PKI cross-certification schemes, WS trust configurations can be easily automated. Moreover, WS facilitate the use of technologies such as proxies, load balancers, and deployment over redundant computer clusters, thus leading to highly dependable infrastructures.

### 6.4.5 Conclusions

In the paper [70], we presented key challenges for identity management in V2X, and proposed design directions of future VPKIs. We presented our Kerberized, standard-compliant VPKI prototype called VeSPA. Our ticket-based multi-service architecture can

satisfy security and privacy needs of an emerging ecosystem of vehicular applications. Additionally, we realize that VPKI architectures can leverage well-defined open standards for Identity Management as in WS. The merging of VC and web technologies can yield numerous advantages. We are investigating further how to instantiate WS-based VPKIs.

## 6.5 Service Oriented Security Architecture

*SEROSA* [73] is a service-oriented security and privacy-preserving architecture for V2X systems. *SEROSA* meets the requirements of authentication, authorization, accountability and user privacy while, at the same time, it offers a comprehensive set of services for resolving the complex challenges in addressing identity management in a multi-service automotive ecosystem. Service discovery and registration support the provision of various personalized services and motivate Service Providers (SPs) to enter the vehicular market while the establishment of trust relations (*federations*), among different entities of the system, facilitates access control across multiple domains. User privacy still remains at the core of *SEROSA*. Towards this, it encompasses existing vehicular communication standards and enhances the underlying Vehicular PKI (VPKI), which is the main building block of current schemes, by leveraging long-term credential and identity managing entities (expected to be deployed in V2X). Novel and efficient authentication protocols, based on the use of Web Services (WS), are proposed to support a multiplicity of diverse services so as to engage the participation of more vehicle operators.

Overall, *SEROSA* extends the current state of the art in V2X by offering: *(i)* diverse service discovery and registration across multiple domains, *(ii)* fine-grained authorization, access control and accountability, *(iii)* user privacy enhancement and service unlinkability, *(iv)* flexible, interoperable, dependable and scalable multi-domain identity management, and *(v)* a full-blown implementation of all system components and protocols, according to the WS paradigm, along with an evaluation of their efficiency, practicality and dependability through extensive and realistic simulations.

### 6.5.1 Adversarial Model

Adversaries in the context of V2X fall into two categories based on their capabilities; *internal* and *external* [74]. Irrespectively of their type, they aim at disrupting the system operation by launching a plethora of attacks. Especially the former, who are authorized participants, might pose as multiple network entities (acting as a Sybil entity). We do not dwell on communication network attacks and outages such as jamming, DoS and DDoS as they are orthogonal to this investigation.

In the context of services, we consider adversarial behavior from users who might either try to access services that are not entitled to (i.e., *free-riders*) or repudiate having received them. Nevertheless, we do not limit adversaries solely to users and additionally address the case of misbehaving authorities. More specifically, we consider the following cases:

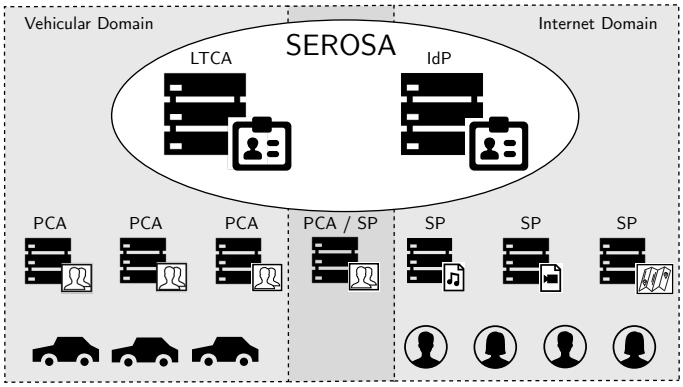



Figure 6.16: Merging V2X with Internet-based services

- *Honest-but-curious SPs*: Authorities that do not deviate from the expected protocol behavior might try to violate the privacy of users by de-anonymizing or profiling them.
- *Fraudulent SPs*: Malicious SPs that might fraudulently accuse users for having received a service in order to benefit from them.

The aforementioned attacks can be launched by single SPs or collaboratively, by multiple *colluding* SPs. We assume that Identity Providers (IdPs) are trusted. Nevertheless, in Sec. 6.5.4 we discuss on mechanisms that can weaken this assumption.

### 6.5.2 Motivation and Design Choices

A gamut of diverse applications and services are expected to find their way to the vehicular ecosystem. Existing Internet-based service providers with multiple security policies and service agreements will be soon offering their services to VC users. Moreover, users seeking personalized services will wish to subscribe to many of them. Furthermore, since vehicle mobility cannot be geographically constrained, it seems likely that such services will span over multiple administrative domains.

We argue that the current notion of V2X security architectures, per-se, cannot address this complex and dynamic setting. On the other hand, a direct application of security solutions currently relevant to the Internet domain is not desired due to the intricacies and rigid security and privacy requirements of vehicular networking environments. What we are seeking is a comprehensive security and privacy-preserving identity management, that emerges as a *synthesis* of the current V2X specific standards with Internet-based services (Figure 6.16). Towards this, we present SEROSA [73], *a service-oriented security and privacy-preserving architecture for V2X* that focuses on the following:

- *Privacy Preserving Identity Management and Authentication*: A service-oriented V2X architecture should provide the necessary means that allow the creation, authentication, and management of the identities of various entities that comprise the

system (i.e., vehicles, authorities and SPs) across multiple domains. However, such an identity management should not come at the expense of user privacy.

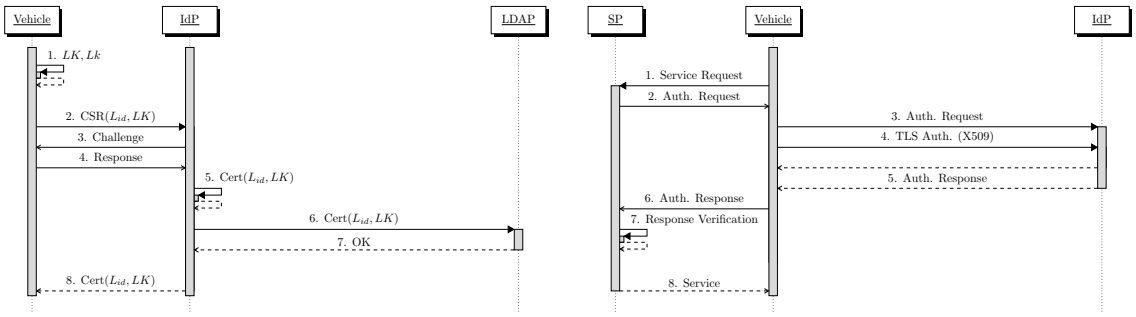
- *Authorization, Access Control and Liability*: Each vehicle should be able to access any service it is entitled to, within any administrative domain in a privacy-preserving, efficient, and accountable manner. Furthermore, it is critical that VPKIs enforce fine-grained security policies to accommodate the requirements of different SPs. Vehicles and system entities should be kept accountable of their actions that could result in system disruption. Proper mechanisms to attribute liability in such cases of misbehavior are essential[1].
- *Service Unlinkability*: Service requests should not be linked and traced back to the long-term identity of originating users.
- *Federated Trust*: A service-oriented V2X architecture should transparently establish strong trust relations (federations) among the different entities of the system. Vehicles receive services from various SPs which in turn rely on multiple IdPs for authentication and access control. This defines the need for a scalable Web of Trust (WoT) between involved stake-holders.
- *Service Discovery*: To support the provision of various personalized services and motivate SPs to enter the vehicular market, a service-oriented V2X architecture should offer the necessary means for facilitating the advertisement and discovery of all the services offered within an administrative domain.

All of the aforementioned functionalities should be provided in a *standard-compliant* and *platform-neutral* manner to ensure interoperability and scalability.

### 6.5.3 System Entities and Design

SEROSA synthesizes Web Services with the credential management entities as they have been defined by the current V2X standards. More specifically, the LTCA (see Figure 6.16) is enhanced to an IdP which offers security services such as Authentication, Authorization and Access Control to any SP. Accordingly, PCAs are SPs that provide standard-compliant pseudonyms to requesting vehicles. Vehicles register to the system and receive certified long-term credentials. Consequently, they query for the services offered within a domain and receive identifiers of the ones they are entitled to acquire. Finally, in case it is mandated by legal authorities, vehicles are evicted from the system (e.g., as a consequence of misbehavior) and prevented from further participation. In what follows we cover the details of all services, offered by SEROSA, and how they can be invoked by system entities. We consider pseudonym provision as a use-case of a service for the rest of the paper.

[1] Misbehavior detection is an important part of vehicular security but orthogonal to this investigation and thus, it is not considered in the work [73].




(a) Vehicle Registration (b) Service Acquisition

Figure 6.17: Registration and service acquisition flow diagrams.

#### 6.5.3.1 Federated Trust

Trust relationships between the entities, comprising SEROSA, are established by means of Security Assertion Markup Language (SAML). In order to establish trust links between the Identity Provider and involved SPs, a WS-Metadata exchange takes place [75]. Metadata are XML-based entity descriptors which contain various pieces of information such as authentication requirements, URI, protocol bindings and digital certificates. More specifically, metadata published by an IdP contain the *X.509* certificates that have to be used by the various SPs in order to decrypt and verify the signatures generated by the IdP. Similarly, SPs publish metadata which contain their corresponding digital identifiers and certificates. After the establishment of a trust relation, SPs can receive identity management services from the IdP.

These mechanisms are used to build large and complex trust models among multiple IdPs and SPs, thus, enabling the establishment of a globally recognized identity management and access control system which spans over multiple V2X domains.

#### 6.5.3.2 Vehicle Registration

The first step of a vehicle's identity life-cycle is its registration to an authority (i.e., IdP) and the subsequent generation of its long-term credentials (Figure 6.17a). The HSM of a vehicle generates a key-pair: a public key $LK$ and a private key $Lk$ (Step 1). Consequently, it issues a Certificate Signing Request (CSR) which contains its long-term identity and $LK$ (Step 2). The IdP then initiates a *proof-of-possession* protocol for verifying the ownership of the corresponding private key $Lk$ (Steps 3 and 4). Upon successful completion, the IdP issues a certificate $Cert_{(L_{\mathsf{id}},LK)}$ and delivers it to the requesting vehicle (Steps 6, 7 and 8). All required information about registered vehicles is stored in a LDAP [76] server. In case the registration process is executed over the network, communications between the vehicle and the IdP are secured over a TLS tunnel (the certificate of the IdP is assumed to be pre-installed on the vehicle's HSM). By the end of this protocol, the vehicle is a legitimate entity of the V2X system and ready to register to any provided services.

#### 6.5.3.3 Service Provision

**Service Registration** In order for the vehicle to be able to receive a service, it first needs to subscribe to the corresponding SP. We assume that trust-relations have already being established between the desired service provider and the IdP at which the vehicle has been registered. To achieve service registration and acquisition we leverage SAML assertions which represent security claims produced by the IdP for the SP. SAML assertions carry the following types of security claims:

- **Authentication Statements:** Assert a SP that the vehicle for which the assertion has been issued, was authenticated according to an agreed authentication protocol.
- **Authorization Statements:** Assert that the vehicle has been deemed eligible for acquiring a service.
- **Attribute Statements:** Information regarding the vehicle attributes such as its type (i.e., public or private vehicles) and its clearance among others.

To register to a service, the following protocol is executed:

$$V \rightarrow SP : request\,\{ses_{\text{id}}, IdP\} \tag{6.33}$$

$$SP \rightarrow V \rightarrow IdP : reg\_req\,\{ses_{\text{id}}, serv_{\text{id}}, SP_{\text{id}}, t\}_{SP_{sig}} \tag{6.34}$$

$$IdP \rightarrow SP : reg\_res\,\{ses_{\text{id}}, serv_{\text{id}}, SP_{\text{id}}, t\}_{IdP_{sig}} \tag{6.35}$$

$$SP \rightarrow V : success\,\{ses_{\text{id}}, OK\} \tag{6.36}$$

Initially, the vehicle ($V$) contacts the desired SP and issues a service registration request. To protect users' privacy, the vehicle does not reveal its $L_{\text{id}}$. A *session identifier* ($ses_{\text{id}}$) is used for identifying and managing the session during further execution of the protocol. This request also contains the $id$ of the IdP to which the vehicle has been subscribed (6.33). Once the SP generates the corresponding registration request, it is relayed by the vehicle to the IdP. Overall, the request contains the $ses_{\text{id}}$, the identifier of the requested service ($serv_{\text{id}}$), the identifier of the SP ($SP_{\text{id}}$) and a time-stamp $t$ (for preventing replay attacks). To ensure authenticity, the request is signed by the issuing $SP$ (6.34). Furthermore, to guarantee confidentiality and execution (of the request) only by the designated IdP, it can be optionally encrypted by means of the key specified during the metadata-exchange between the SP and the IdP (Sec. 6.5.3.1). Additional pieces of information such as billing can also be included in the request. Upon reception, the IdP authenticates the vehicle on the basis of its *long-term identity*. If successful, it issues a registration response as a proof that the vehicle has been registered for the service (6.35). Finally, the SP sends an acknowledgment back to the vehicle (6.36).

**Service Discovery** A vehicle that is within a foreign administrative domain and wishes to discover what services are being offered within it, can issue a Simple Object Access Protocol (SOAP) request to a (Web Services Discovery Language) WSDL enabled server and receive a description of them [75]. For instance, if the vehicle wishes to discover PCA services, then it can query the server for them. In the case that trust relationships have been established between the discovered PCA and the IdP (responsible for the domain from which the vehicle originates from), the vehicle can receive the required pseudonyms. The details of service acquisition from foreign administrative domains is described later in this section.

**Service Acquisition** Figure 6.17b illustrates the steps executed during the authentication protocol [2]. Initially, the vehicle requests for the desired service (in an anonymous way) without disclosing its $L_{\text{id}}$ to the SP (Step 1). Thereupon, the service provider issues an *authentication request* designated for the IdP. According to the specifications of SAML [75], the request is relayed by the vehicle (Steps 2 and 3). Consequently, the vehicle engages in an interactive authentication protocol with the IdP (based on TLS) by means of their digital certificates. It reveals its long-term identity and the IdP examines if it is entitled to receive the requested service (Step 4). Upon successful authentication, the IdP issues an *authentication response* which contains a SAML assertion. This assertion does not reveal the $L_{\text{id}}$ of the vehicle and instead uses a *transient identifier* $tr_{\text{id}}$ (a random identifier generated by the IdP). Such identifiers are used to obfuscate the long-term identity of the requesting vehicle in order to prevent SPs from linking service requests to it. They are valid only for a single authentication request and change for each subsequent attempt. Upon reception, the SP validates the authentication response and examines the eligibility of the vehicle with respect to the service; accordingly it grants or denies access (Steps 7 and 8). The complete protocol flow is performed over HTTP. To guarantee the authenticity of both the IdP and the SP, we leverage one-way TLS authentication that additionally ensures the confidentiality and integrity of communications.

*Use-Case*: Regarding pseudonym acquisition, the PCA serves as a service provider which issues pseudonyms to vehicles according to the $1609.2$ specification [56]. Similar to the registration phase, the public ($PK_i$) and private ($Pk_i$) keys of the pseudonym are generated inside an assumed HSM. Once a vehicle is authenticated (following the above described protocol), it can request pseudonyms from the PCA. A certified pseudonym $Ps_i$ is a digital signature, produced by the PCA, over the public key $PK_i$.

An advantage of our scheme is that there is no need to initiate the authentication process whenever the vehicle wishes to request a service. Web services allow SEROSA to support Single-Sign-On (SSO) capabilities meaning that SAML assertions can be reused transparently for requesting services from multiple SPs, residing within federated domains. SSO can accelerate service reception in case no cellular network-based connectivity is available; thus, V2I communications are only feasible when the car is within the proximity of RSU. Nonetheless, receiving services in this manner might harm the privacy

[2] Due to space limitations we omit the specifications of SAML and instead refer the reader to [75].

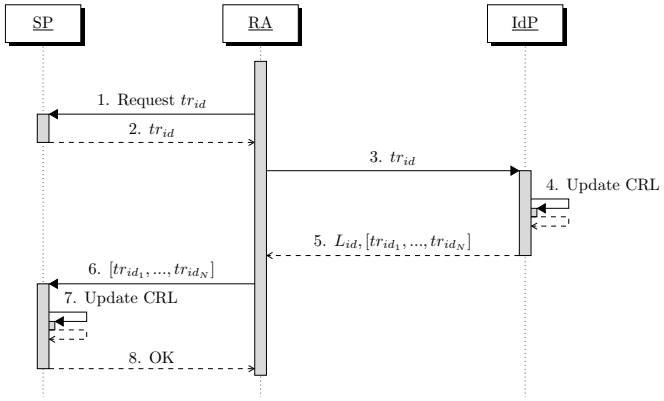



Figure 6.18: Pseudonym resolution and revocation

of vehicle operators. A detailed discussion on SSO and its impact on user privacy can be found in Sec. 6.5.4.

SEROSA allows registered vehicles (within one administrative domain) to be authenticated at foreign domains by means of SAML assertions. To enable this scenario we make use of *delegated authentication*. More specifically, when a vehicle (registered with domain $D_A$) accesses a SP within a foreign domain ($D_B$), it is redirected for authentication to $IdP_B$ of $D_B$. Since $IdP_B$ has no information regarding the vehicle, it redirects the request to the $IdP_A$ of $D_A$. Consequently the authentication protocol is executed and a SAML assertion is generated by $IdP_A$ and endorsed by $IdP_B$. This assertion is presented to the SP who eventually delivers the service.

#### 6.5.3.4 Pseudonym Resolution and Revocation

For attribution of liability, our architecture provides mechanisms that allow tracing of a pseudonym $Ps_i$ back to the vehicle's long-term identifier $L_{\text{id}}$. In this context, authorities might additionally request the eviction of misbehaving vehicles from the system. To achieve resolution of $Ps_i$, our scheme assumes a Resolution Authority (RA), that could be any law enforcement agency, which initiates the revocation process. An illustration of the protocol steps are presented in Figure 6.18.

The RA requests from the PCA the identifier $tr_{\text{id}}$ of the SAML assertion for which $Ps_i$ was issued (Step 1). Consequently, the PCA responds with the corresponding $tr_{\text{id}}$ (Step 2). The RA then provides the IdP with the $tr_{\text{id}}$ that generated $Ps_i$ (Step 3). The IdP updates the Certificate Revocation List (CRL) to include the $Cert_{(L_{\text{id}},LK)}$ of the vehicle with the corresponding transient identifier (Step 4). From this point on, the misbehaving vehicle cannot be authenticated and, thus, receive new pseudonyms. Additionally, the IdP provides the RA with the $L_{\text{id}}$ (of the vehicle) and the list of all the $tr_{\text{id}_j}$ for which it has issued assertions (Step 5). This list is dispatched to the PCA which in turn updates its CRL to include all the $Ps_i$ issued under these transient identifiers.

Steps $1, 2, 3$ and $5$ of the protocol suffice in case simple pseudonym resolution, and not complete eviction from the system, is requested.

### 6.5.4 Security and Privacy Analysis

In this section, we qualitatively analyze the security and privacy properties of SEROSA with respect to the requirements presented in Sec. 6.5.2. The *integrity* and *confidentiality* of all Vehicle-to-Infrastructure communications are protected by means of secure and authenticated (i.e., TLS) channels so that the system is immune to a wide range of attacks, e.g., session hi-jacking, eavesdropping, etc. In this context, the IdP is responsible for strict identification and authentication of all vehicles while authorities are authenticated through the use of digital certificates.

As discussed above, SAML offers great versatility when it comes to *Access Control and Policy Enforcement*. Identity and service providers can exchange authorization information inside SAML assertions serving as the Policy Decision Points and Policy Enforcement Points, respectively. Additionally, Role Based Access Control is feasible on the basis of SAML attribute statements, which can associate vehicles to specific roles (i.e., public or private). In case of more complex security policies, over multiple domains, our scheme supports eXtensible Access Control Markup Language [77]. Furthermore, the revocation protocol described in Sec. 6.5.3.4 can be used to revoke the anonymity of vehicles when they have been deemed misbehaving (*liability attribution*). All of the above provide a globally recognized identity management and access control system which can span over multiple VC domains.

Compounding the issue of *Sybil attacks*, SEROSA mitigates their effects by means of HSMs similar to the ones currently developed by PRESERVE [67]. More specifically, these trusted platforms serve as secure storage for all produced cryptographic keys. Since both the $L_{\text{id}}$ and $Ps_i$ of vehicles are bound to such keys (that never leave the HSM), Sybil attacks do not pose any threat to the system.

Our architecture also ensures privacy in the case of "honest-but-curious" and colluding infrastructure. Curious SPs may attempt to passively violate a vehicle's privacy profile. However, linkage of subsequent service requests (originating from the same vehicles) is infeasible since for each SAML assertion a different transient-identifier is used. Nevertheless, if such SPs collude with an IdP, assertions issued by the misbehaving IdP can be tracked and resolved to the $L_{\text{id}}$ of the vehicle. To provide resilience even in this scenario, anonymous authentication methods like group signatures [78] or cryptographic vehicular tokens [79] could be used. At the same time, unauthorized users that have not subscribed to a service (see Sec. 6.5.3.3), cannot fraudulently access it as the involved IdP will not issue a SAML assertion (during authentication). This point renders SEROSA secure against *free-riders* (under the condition that the IdP does not misbehave). Moreover, SAML tokens serve as proof-of-service receptions and, thus, malicious users cannot repudiate having received a service.

Re-usage of previously acquired assertions in a SSO manner may lead to the linkage of subsequent service requests and, thus, might harm the privacy of vehicle operators. This

| | IdP | PCA | RA |
|---|---|---|---|
| Virtual Machines | 1 | 3 | 1 |
| Dual-core CPU 2.0 GHz | 8x | 1x | 1x |
| System Memory | 4 Gb | 2 Gb | 1 Gb |
| Web Service Software | SimpleSAMLphp | Shibboleth 2 | ✗ |
| Apache Web Server | ✓ | ✓ | ✓ |
| Apache Load Balancer | ✗ | ✓ | ✗ |
| MySQL Database Server | ✓ | ✓ | ✗ |
| OpenLDAP Server | ✓ | ✗ | ✗ |
| OpenSSL | ✓ | ✓ | ✓ |

Table 6.4: The host setup for the system deployment.

trade-off needs to be addressed by *policies* which (based on the mobility status of the vehicle) will define if and how SSO will be used.

### 6.5.5 Performance Evaluation

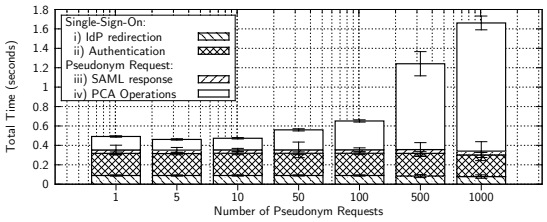
In this section, we evaluate various aspects of SEROSA focusing on the efficiency and reliability of the proposed architecture. Three properties are of interest namely *vehicle authentication*, *pseudonym acquisition* and *revocation*. In an attempt to avoid confining our results, we have also considered the *network latency* induced by vehicular mobility; this usually implies volatile network connectivity. In all cases, our goal is to provide strong evidence with respect to the feasibility of the proposed approach under the requirement of efficient service provision in a vehicular networking environment.

#### 6.5.5.1 Evaluation Environment Setup

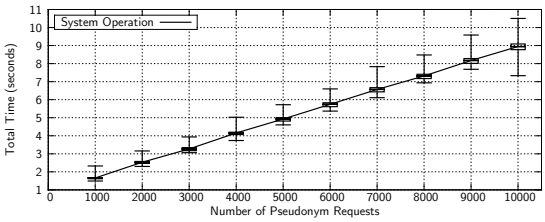
In all our experiments, we considered a testbed comprising various Virtual Machines, each one dedicated to a different authority (IdP, PCA, and RA), together with remote clients that request to access some of the provided services. We employed the OpenSSL library for cryptographic operations, such as ECDSA, RSA signature schemes, and TLS connection establishment. To abide by the current standards and directives, we required that the ECDSA keys are computed over curves of $256$-bit primes. Furthermore, for emulating the network delay, we employed a queuing discipline that increases randomly the data link delay following a normal distribution with $\mu = 10\,ms$ and $\sigma^2 = 2.5$. A summary of the whole system hardware and software configuration can be found in Table 6.4.

#### 6.5.5.2 Pseudonym Request

A critical aspect for any VPKI is its *scalability*; while the $L_{\text{id}}$ has (by definition) low variation frequency (from months to years), pseudonyms need to be frequently updated within time intervals that could vary from minutes to days, depending on the system policies.




(a) $t_i$ for a single request containing an exponentially increasing number of pseudonyms, averaged over $1\,000$ runs.




(b) $t_{\text{TOT}}$ with respect the number of pseudonyms in a single request. Samples from $1\,000$ observations.

Figure 6.19: Performance evaluation for pseudonym requests.

Consequently, a PCA should be able to handle not only as many requests per second as possible, but also support high-level concurrency typical to a dense vehicular mobility scenario. For this reason, we replicated the functionalities of a PCA with $2$ web-servers behind a proxy/load-balancer. We evaluated the time required for a single pseudonym request and studied the impact of multiple concurrent requests through a real-case scenario.

As described in Sec. 6.5.3.3, to obtain the pseudonyms a vehicle needs to: (i) contact the PCA and receive the SAML authentication request, (ii) authenticate itself to the IdP, (iii) provide the assertion back to the PCA and, finally, (iv) send the pseudonym signing requests in order to receive the new pseudonyms. We assume that the pseudonym signing requests are prepared in an off-line manner (by the vehicle) before protocol instantiation. We also consider the time for each authentication step $t_i$, $t_{ii}$, $t_{iii}$, and $t_{iv}$ while $t_{\text{TOT}} = t_i + t_{ii} + t_{iii} + t_{iv}$ denotes the total pseudonym acquisition time.

**Single Vehicle** Each step of the service acquisition protocol ($t_i$) has been sampled $1\,000$ times for varying numbers of requested pseudonyms (from $1$ to $1\,000$ assuming an exponential increment). The results are depicted in Figure 6.19a. As expected, the first $3$ steps do not depend on the size of the request, while step (iv) shows the correlation with the number of requested pseudonyms. A significant increase in terms of latency is observed only for requests containing more than $10$ pseudonyms. Indeed, before reaching this point, the network operation time compensates the time required for signatures meaning that the processing time is negligible with respect to the network latency.

Moreover, $t_i + t_{ii}$ is the time required for a vehicle to receive a valid SAML assertion. The assertion could be reused in subsequent requests (Sec. 6.5.3.3), avoiding an overhead of approximately $300\,ms$.

In case of vehicle mobility adding time constraints (due to RSU coverage) and *no-refilling* pseudonyms while the vehicle is parked [80], the latency for pseudonym requests of $1\,000$ pseudonyms ($t_{\text{TOT}} \simeq 1.62\,s$), can be easily accommodated.

Moving on to total time measurements, Figure 6.19b depicts measured $t_{\text{TOT}}$ sampled over $1\,000$ observations assuming a linear increase of the size of the request, from $1\,000$ to $10\,000$ pseudonyms per request. We do not assume any limitation on the request size since a vehicle could (theoretically) refill a rather large number of pseudonyms, i.e., before a long trip. Therefore, the system should be able to handle any request size without any significant performance degradation. As the figure shows, the latency grows linearly with the number of pseudonyms contained in a single request and is around $9\,s$ for $10\,000$ certificate generations (without any hardware acceleration).

Finally, our system outperforms the scheme presented in [66]: without considering the request preparation and verification of the received $Ps_i$ (since both can be performed asynchronously), VeSPA requires approximatively $5\,s$ more compared to SEROSA for requests of $1\,000$ $Ps_i$.

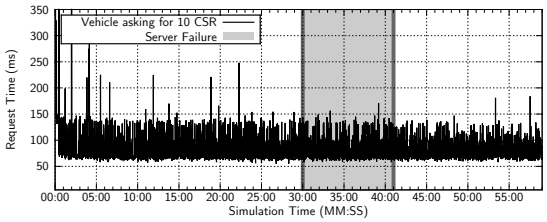
**Real-case scenario** To evaluate the efficiency and scalability of SEROSA in a dense urban environment, we accumulated data from a real life scenario. We extracted $5\,000$ vehicular traces within the city of Cologne (Germany) from the "TAPAS Cologne" project [81]. In order to emulate the vehicles, we assigned a thread to each one of the traces and assumed a pseudonym request policy of $10$ pseudonyms every $10$ minutes (i.e. a lifetime of $1$ minute for each [82]).

Figure 6.20a depicts the observed latency for a simulation interval of $1$ hour. As it can be seen, the system response time is around $100\,ms$ (on average). During the $1$ hour TAPAS scenario simulation, we also introduced a temporary outage of the PCA by disconnecting completely one of the two Web-servers behind the load-balancer. As shown in the shaded area, the request latency does not increase and the PCA recovers transparently from such an operation disruption.

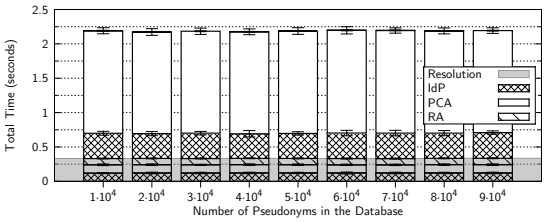
#### 6.5.5.3 Resolution and Revocation

In this section, we conduct additional simulations to further examine the behavior of SEROSA in the case of pseudonym resolution and revocation. As described in Sec. 6.5.3.4, the resolution and revocation protocols can be summarized as follows: (i) RA inquires the PCA for the $tr_{\text{id}}$ that was used by the vehicle to request pseudonym $Ps_i$, (ii) RA asks IdP to revoke the $L_{\text{id}}$ associated with this $tr_{\text{id}}$ and to provide all the other issued $tr_{\text{id}_i}$, (iii) RA finally requests PCA to revoke all the pseudonyms associated with all the $tr_{\text{id}_i}$. Therefore, the property of interest is the time spent on PCA ($t_{\text{PCA}}$), IdP ($t_{\text{IdP}}$), and RA ($t_{\text{RA}}$) for a single pseudonym resolution and revocation process with respect to the size of the pseudonym set.

To maximize the entropy of the pseudonym set, we require that pseudonyms are assigned to different $L_{\text{id}}$ with equal probability. Towards this direction, we assume that each vehicle ($L_{\text{id}}$) uses $10$ different ($tr_{\text{id}}$) to request $10$ pseudonyms. Therefore the overall ratio is ($1\,L_{\text{id}}$ : $10\,tr_{\text{id}}$ : $100\,Ps_i$).




(a) $5\,000$ emulated vehicles operating for $1$ hour and asking for $10$ pseudonyms every $10$ minutes.




(b) Time spent on each system component for a single pseudonym revocation. The gray area denotes the time spent for the resolution of the revoked pseudonym.

Figure 6.20b summarizes the latency of each system component for a single pseudonym revocation. Results are averaged over $100$ runs, with the set of revoked pseudonyms (in the database) increasing from $10\,000$ to $100\,000$ items linearly. As it can be seen, the performance of the revocation protocol is not affected (at all) by the number of already revoked pseudonyms. Moreover, as expected, it holds that $t_{\mathsf{PCA}} > t_{\mathsf{IdP}} > t_{\mathsf{RA}}$, due to the fact that RA needs only to dispatch the commands to the IdP and the PCA.

Additionally, the time required for resolving the inquired pseudonym, i.e., requesting the $tr_{\mathsf{id}}$ from the PCA and the $L_{\mathsf{id}}$ from IdP, is about $320\,ms$. In [83] the authors evaluated the pseudonym resolution performance of their system (under similar conditions) and demonstrated a latency of $550\,ms$ which increases with the size of revoked pseudonyms database.

On the other hand, both the IdP and the PCA must execute the actual revocation for each certificate, i.e. update and sign the Certificate Revocation List. However, their lists differ by one order of magnitude. If such a delay is considered critical, an Online Certificate Status Protocol (OCSP) could be employed to reduce the resulting communication overhead.

### 6.5.6 Conclusions and Future Work

In [73] we presented SEROSA a secure and privacy-preserving service-oriented architecture for V2X. SEROSA provides a comprehensive set of identity management and access control services while still guaranteeing the security and privacy requirements of V2X. Leveraging widely accepted telecommunication standards, such as Web Services, SEROSA transparently accommodates the needs of vehicles and service providers irrespectively of their location or the V2X domain they belong to. Through extensive evaluations we demonstrate its dependability and efficiency compared to state of the art VPKIs.

The merging of vehicular networks and web technologies can yield numerous advantages for V2X. Furthermore, the extendability of WS leaves space for anonymous and unlinkable authentication schemes that can ensure privacy even in the presence of trusted-but-curious infrastructure and, thus, reduce the knowledge gathered by the IdP.

# 7 Security architecture

## 7.1 Secure Storage of Private Keys

### 7.1.1 Introduction

Vehicle-to-Vehicle (V2V) and Vehicle-to-Infrastructure (V2X) communication will enable vehicles to exchange information regarding safety, traffic condition and infotainment with each other. The On-Board Unit (OBU) will play a key role in these systems, as it manages incoming/outgoing messages and also performs security and privacy functions. Security and privacy of V2X communications are mandatory to enable a successful deployment of Intelligent Transportation System (ITS). As many V2X applications have a potential impact on traffic safety, their communication must be secured for obvious reasons [84]. Only messages from authenticated vehicles should be processed by receiving vehicles to prevent, for example, false safety-related warnings. Current standardization efforts, both in the U.S. and Europe, foresee that ITS will require the establishment of a Public Key Infrastructure (PKI), which manages trust and certificates in the ITS. The current set of standards [1, 85, 86] mandates the use of Elliptic Curve Digital Signature Algorithm (ECDSA) with P-256 elliptic curve for message authentication. A naïve implementation of authentication mechanisms breaks user privacy as every receiver learns the identity of the sender. Therefore, a pseudonymous credential – short *pseudonym*– should be implemented in order to prevent authentication to facilitate direct vehicle identification. One single pseudonym is not enough to ensure a sufficient level of privacy. Instead, this pseudonym has to be changed frequently, and even then, a powerful attacker may be able to track vehicles [87]. A central question here is how many such pseudonyms a vehicle possesses and how often it would have to contact the PKI for renewal. In general, a frequent connection to the PKI to renew pseudonyms cannot be guaranteed because large-scale coverage by road-side units (RSU) or cellular communication in every vehicle is considered unrealistic during early years of V2X deployment. Thus, the OBU has to store a potentially large set of pseudonyms to allow frequent change of pseudonyms in absence of backend connectivity. In a worst case, a vehicle would only be able to load new pseudonyms during (bi-)annual inspections in a garage. Recent research estimates that an OBU is required to store 105,120 pseudonyms for one year with each pseudonym valid for five minutes [88].

Each of those pseudonyms consists of a public-private key pair and a corresponding certificate and especially the private key needs to be stored securely to not compromise security of the overall system. If an attacker acquires access to the secret keys stored in a vehicle, she could perform sybil attacks, spoofing attacks, and in general jeopardize the

authentication and privacy of the victim. In consequence, it must be guaranteed that a private key is strictly secured during all events in its life cycle. This goal can be achieved by designing systems to securely create, manage and destroy (private) keys, maintaining an audit trail of every operation executed during their existence. Hardware Security Modules (HSMs) [89] are specifically designed to protect private keys. HSMs are specialized tamper-proof devices in which cryptographic functions and embedded software properly manage keys and control their life cycles. They are designed in such a way that if an unauthorised attempt to access them is made, this is considered an attempt to tamper and all critical internal parameters and keys are destroyed. The HSM features make them a crucial component in automotive platform security [90, 91].

However, HSMs are especially expensive if implemented on an FPGA [92], and a secure storage within an HSM adds complexity to the overall system. With ECDSA P-256 curve, the private keys of the one-year pseudonyms set proposed in [88] would require $256$ bits $\times$ $105,120 = 3.2$ Mbytes of secure storage – not considering yet any overhead for data management. This requirement is too high as current solutions offer a maximum of 512 kbytes [93, 94]. Therefore, we aim at trading secure storage of cryptographic key material for regular storage (i.e. outside of the HSM).

### 7.1.2 System Model

#### 7.1.2.1 Physical Unclonable Functions

A Physical Unclonable Function (PUF), as introduced in [95, 96], is a primitive that is bound to a physical system and extracts a pseudorandom bit string for key generation by mapping a set of challenges $C_i$ to a set of responses $R_i$. This challenge-response behavior is highly dependent on the physical properties of the device in which the PUF is contained or embedded. PUFs consist of two parts:

i) a physical part, which is an intractably complex physical system that is very difficult to clone. It inherits its unclonability from uncontrollable process variations during manufacturing. For PUFs on an Integrated Circuit (IC), these process variations are typically deep-sub-micron variations such as doping variations in transistors.

ii) an operational part, which corresponds to the function.

In order to turn the physical system into a *function*, a set of challenges $C_i$ (stimuli) has to be available to which the system responds with a set of sufficiently different responses $R_i$. The function can only be evaluated using the physical system and is unique for each physical instance because of process variations. Moreover, it is unpredictable even for an attacker with physical access.

PUF responses are noisy by nature. This means, that two calls to a single PUF with the same challenge $c_i$ will output two different but closely related responses $r_i, r_i'$. The measure of closeness can be defined via a distance function, e.g., the Hamming distance. This distance function should be small for responses from the same device and very large for PUF responses from different devices. Since the plain PUF responses are noisy, they

cannot be used as a key. In order to derive reliable and uniform data from (imperfect) sources of randomness, such as a PUF, the concept of a fuzzy extractor [97] or helper data algorithm [98] was introduced. Thus, we obtain a *master secret* from the fuzzy extractor. This master secret can be the seed for a key generation scheme to derive public/private key pair(s) which can then be used as a pseudonym(s). Alternatively the master secret can be used to first seed a key derivation scheme, which results in a larger amount of data that can then be used as seeds for key generation processes.

The formulation of abstract properties of PUF types as well as the development of PUF constructions are still a matter of active research [99]. In this paper we use the terminology proposed by Rührmair et al. [100] and refer to Strong PUFs[1] and Weak PUFs[2]. To the best of our knowledge, no research has investigated the applicability of PUFs for storage of large numbers of private keys (or keypairs) as required by the V2X pseudonym scenario.

#### 7.1.2.2 On-Board Unit Architecture

Figure 7.1 shows the current ETSI reference architecture of an On-Board Unit (named "*ITS Station*" in the standard). It shows the different layers and particularly the security layer. One can notice the Hardware Security Module (HSM) within. As Figure 7.1 is an abstract view of an OBU, and thus, does not represent the hardware, Figure 7.2 shows a simplified hardware architecture. An OBU includes CPU, host memory (RAM), regular storage, and an HSM. For simplicity, we represent an HSM that only includes a true random number generator (TRNG), cryptographic primitives (AES, ECC), secure storage, and a PUF. However, one should notice that the PUF could be outside of the HSM (represented in dashed line in Figure 7.2).

Indeed, the PUF could be fully integrated in the CPU, GPU or RAM [101], but also attached to the OBU as an external device. In the remainder of this paper, we consider the PUF as an external device as we compare against classic secure storage solutions (e.g. smart card, secure token), which are mostly externally attached to the OBU. A consequence of being outside the HSM is the lack of a secure computation environment. An attacker (described in Section 7.1.2.3) could then access to the memory to steal key material. However, this drawback is limited by the limited lifetime of the pseudonyms (i.e. certificate). We further discuss the issue of secure computation in Section 7.1.5.2.

#### 7.1.2.3 Attacker Model

With respect to secure storage we consider attackers who want to access the content that is placed in the secure storage container. In our context, the aim of the attacker is to copy the private key material used as pseudonym of a vehicle. We differentiate between two attacker goals: An attacker might try to get access to the private keys for the currently

[1] Labeled as "minimum readout time" PUF (MRT-PUF) [99]
[2] Also known as Physical Obfuscated Keys (POKs)

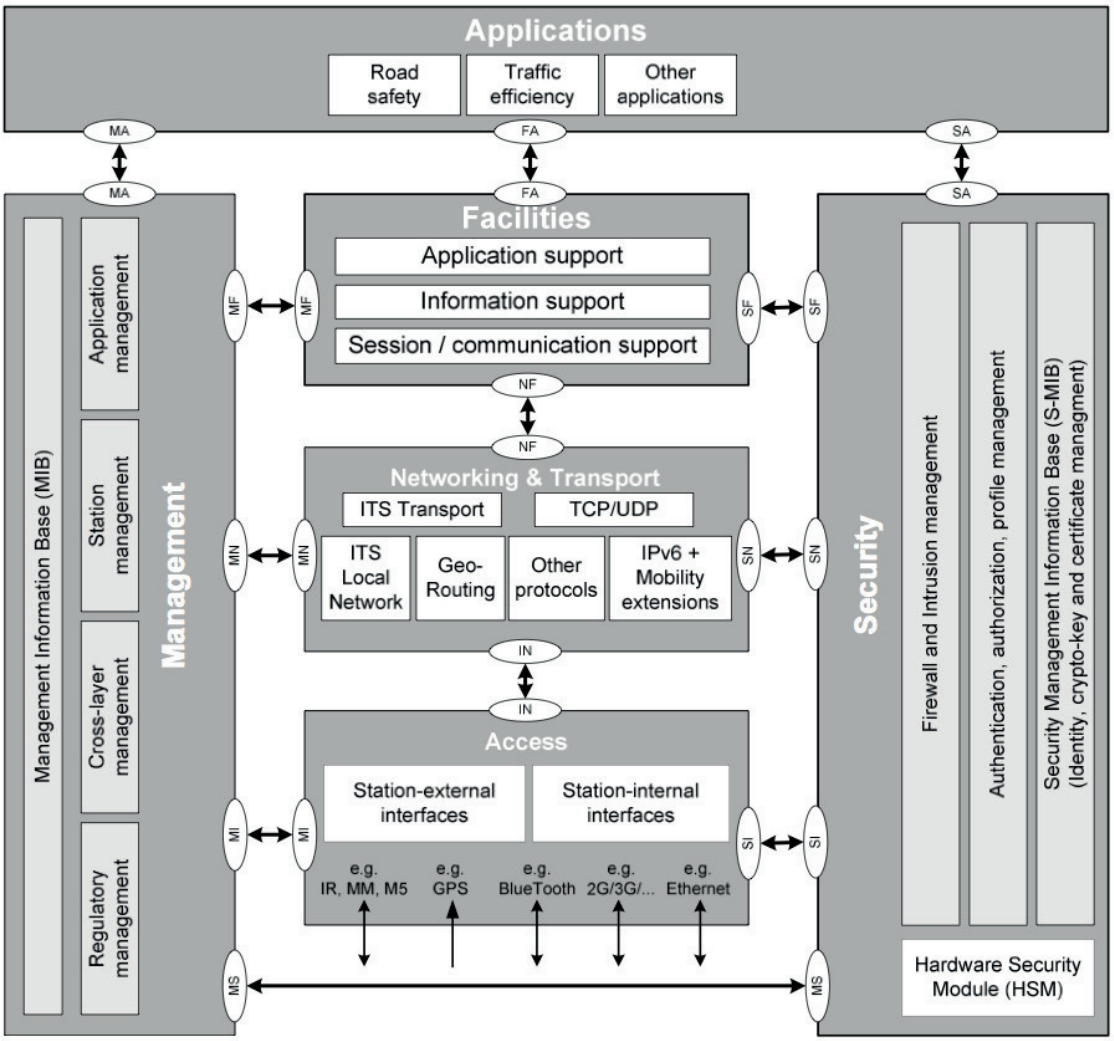



Figure 7.1: ETSI architecture of an OBU [1]

used pseudonym or the attacker might aim to access all private keys for all pseudonyms provisioned in the OBU.

An attack against the OBU can be performed by injecting a payload into the system, which would trigger malicious actions. Since the OBU does not provide a user interface, such a payload needs to be injected into the system remotely. OBUs offer a number of opportunities to an attacker to input data into the system remotely. Most notably the networking and communication applications in the OBU are processing data from external sources, which might be controlled by an attacker. Exploitation of security holes in these applications can lead to different levels of access to the contents of the OBU:

1. Access to filesystem data

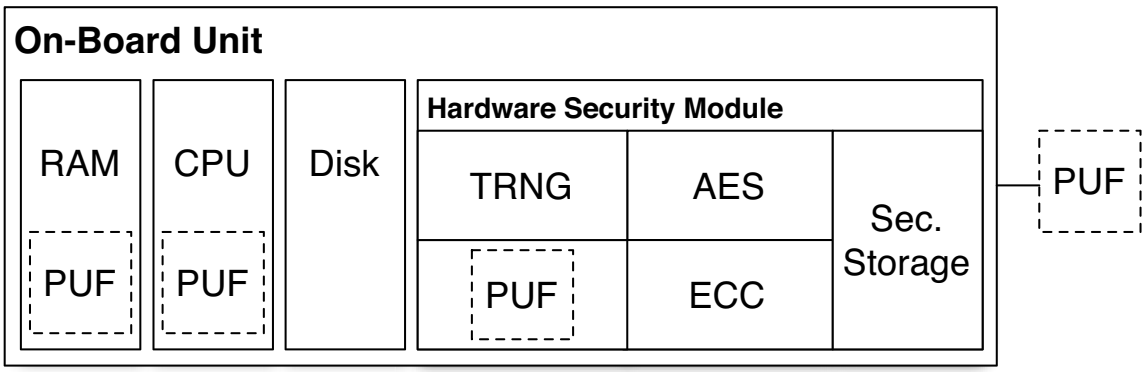



Figure 7.2: Simplified hardware architecture of an OBU

2. Access to application memory
3. Access to hardware devices and code execution

We consider attackers with an escalating set of capabilities to evaluate the level of protection offered by the different proposed techniques. Access to filesystem data serves as a baseline scenario to illustrate that the basic secure storage mechanisms work. An attacker should never be able to access key material based on filesystem access. The second level of access represents more severe information disclosure attacks. In a scenario without secure computation this will allow an attacker to extract key material that is currently in use. A third type of attacker has the ability to execute arbitrary code on the OBU, and thus, is able to arbitrarily interact with any device attached to the OBU. For an external device, such an attacker is indistinguishable from a regular host application. Nevertheless, we consider this type of attacker as the most powerful type of attacker, because she has full control of the OBU.

In this paper we do not investigate hardware attacks against the secure storage. The intrinsic tamper-resistance of PUFs is assumed to protect against this kind of attacker. We assume equally that the tamper-proof enclosure of classic secure storage solutions is effective.

### 7.1.3 Classic secure storage

In this section we propose ways to implement efficient secure storage of large numbers of private keys for use in secure pseudonymous communication. We differentiate between regular storage and secure storage requirements for keys and related support data. The proposed solutions have different space requirements to store and protect these data, which will be our main metric to compare the efficiency of the proposed methods. As a baseline, we assume the availability of classic external secure storage, for example in the form of a physical *smart card* or as part of a dedicated secure *storage token* on a USB device. Our goal is to minimize the usage of this resource or eliminate the use of this resource entirely.

#### 7.1.3.1 Individual key storage

The canonical way to handle secure storage is to assume the presence of a dedicated device, which is isolated from the host. The security attributes of this solution are derived from the fact that the memory on this type of device is only accessible through a well-defined security API.

No other way should exist to access the data, neither in software nor in hardware. The protection against hardware access is usually achieved through protective tamper-proof enclosures or self-destructive coating. The details of the hardware and the communication protocol as well as options to perform secure computation on the device are out of the scope of this paper.


Figure 7.3: All keys in secure storage

Figure 7.3 illustrates the fact that all $n$ keys need to be stored in the secure external device. The limiting factor of this solution is the raw amount of data that needs to be stored in this scheme. As introduced in Section 7.1.1, it is expected that secure pseudonymous communication in vehicular networks will require multiple megabytes of private keys. The key management and the amount of secured data storage increase the cost of such a solution as the number of pseudonyms grows.

#### 7.1.3.2 Encrypted storage

Storing private keys in encrypted form in regular storage, e.g. in a file or database, is a common solution found in password management software for consumers. This kind of solution is usually tied to a master password and a password-based key derivation function to decrypt the data structure. For non-interactive use, we can adapt this solution to use a master key stored inside a secure storage device to encrypt and decrypt the private keys as needed. Figure 7.4 illustrates this method. Using a master key with sufficient entropy in a secure data store allows us to avoid key stretching techniques [102] that are typically employed in password based key derivation functions like PBKDF2 [103], bcrypt [104], or scrypt [105].

The advantage of this method compared to a classic secure storage solution (Section 7.1.3.1) is that only one master secret is required to be stored securely. This master secret will subsequently unlock any number of additional private keys, which can be stored in encrypted form in regular unsecured memory. Conversely, the disadvantage is of course that now an attacker only requires this master key and the encrypted–but not securely stored–data

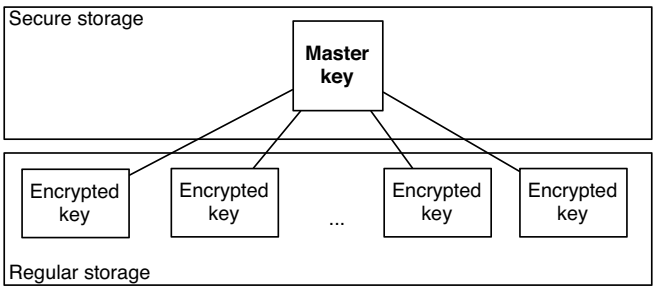



Figure 7.4: Keys retrieved from encrypted file in regular storage using a securely stored master key

structure of private keys to not just compromise one private key, but all private keys stored in this data structure.

#### 7.1.3.3 Key derivation

Taking the concept of using a master secret even further, we use a key derivation function to derive secret keys from the master secret. A practical implementation of this idea uses a keyed pseudo-random function to derive a sequence of bits from a single master key (or seed). These bits can be used as a secret key for symmetric cryptography, but also as a deterministic source of random bits in the generation process of an asymmetric ECDSA key pair [106]. Figure 7.5 illustrates this abstract process. Well known constructions of such key derivation functions include KDF2 [107–109], HKDF [110–112], and the set of deterministic random bit generators (without reseeding) recommended by NIST [113].

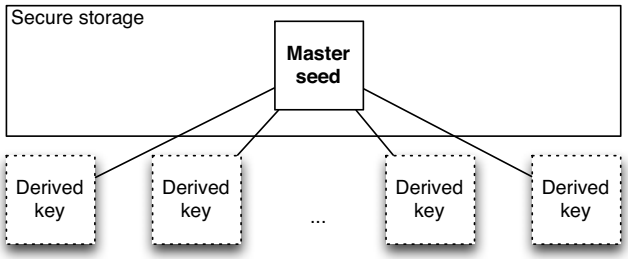



Figure 7.5: Keys regenerated through a key derivation function using a securely stored master key

An additional advantage of using key derivation functions is the reduction of the communication overhead. Indeed, if the CA generates and stores the master seed for the vehicles, it is no longer necessary to submit the key pair through a secure communication channel to the vehicle. It is enough to transfer only the certificates, which needs to include context information, to allow the vehicle to derivate the matching key pair independently. These

information do not even require protection, enabling the use of unauthenticated broadcast channels or public certificate servers for the delivery of new pseudonym certificates.

### 7.1.4 PUF-based secure storage

In this section we propose secure key storage solutions which do not rely on any classic external secure storage, but instead, use Physical Unclonable Functions (PUFs) to achieve the desired security. As introduced in Section 7.1.2.1, we consider two types of PUFs: Strong PUFs and Weak PUFs.

#### 7.1.4.1 Strong PUF-based secure storage

Ongoing research on applications of PUFs for key generation and regeneration is focusing on the fuzzy extraction algorithm. From an application perspective in the vehicular communication context, we observe that we need to securely store large numbers of secret keys. Our proposal, which is summarized in Figure 7.6, requires the use of a Strong PUF [100] that fulfills the following requirements:

1. It must be impossible to physically clone the PUF.
2. A complete determination/measurement of all challenge-response pairs (CRPs) within a limited time frame (such as several days or even weeks) must be impossible.
3. It must be practically impossible to numerically predict the response to a randomly selected challenge, even if many other CRPs are known.

These requirements were setup by Rührmair et al. with scenarios in mind that require a large number of interactive challenge-response cycles, e.g., for remote authentication. Attackers could, e.g., send specific challenges to the PUF, record the responses, and then try to perform a so called “model building attack” [100]. For our usage of PUFs for pseudonym storage, an attacker will not be able to directly query the PUF and see the responses. Only the CA is supposed to be able to communicate with the OBU, and PUF responses will only be used to derive key pairs from it. This effectively removes the unconditional need for requirement 2, although for cost effectiveness of this solution it is still desirable to demand a large space of challenge-response pairs (CRPs). In Section 7.1.4.2, we propose an alternative solution that can tolerate the availability of only small amount of CRPs per PUF (Weak PUF).

The idea that we pursue in this proposal is to derive key material from PUF responses. The use of a Strong PUF implies that we have a large space of challenge-response pairs, which enables us to derive large numbers of keys. As in the solution based on KDF, we use deterministic random bits as a source of entropy in the key generation process of asymmetric ECDSA key pairs [106].

The amount of input data required to generate a stable amount of responses is highly dependent on the attributes of a concrete PUF construction. In general we require a set of

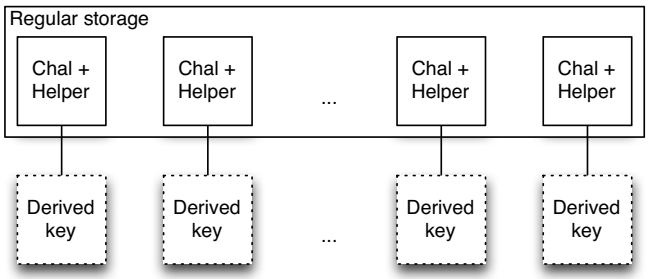



Figure 7.6: Keys reconstructed securely from a strong PUF using regularly stored challenges and helper data

chosen challenges and a set of helper data, which is generated by the fuzzy extractor during the initial key generation process. Depending on the type of PUF construction, a total amount of $n$ challenges $c_n$ of $x$ bits length is required to generate $m$ bits of output. These $m$ bits of output then need to be stabilized using a fuzzy extractor (see Section 7.1.2.1). In the initial key generation process the fuzzy extractor will generate helper data. In subsequent calls to the PUF, this helper data is instead used by the fuzzy extractor to reconstruct the same stable response. In both cases, the fuzzy extractor will consume a percentage of the data for entropy compression and error correction. The factor of the data reduction $r$ as well as the length $y$ of the helper data $W$ depends on the type and configuration of the fuzzy extractor. The configuration needs to be calibrated based on the expected *error probability* and *entropy quality* of a given PUF construction.

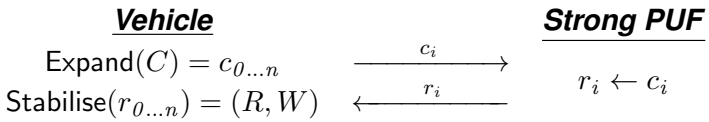



Figure 7.7: An initial challenge ($C$) gets expanded into $n$ challenges ($c_i$), which generate responses ($r_i$) in the PUF. The vehicle combines these into a final response ($R$) and helper data ($W$).

For an overall amount of stable response bits $z$, we can calculate the number of required challenges as $n = \frac{z}{m \cdot r}$. To enable reconstruction of stable responses, we would need to store the $n$ challenges of size $x$ and the helper data $W$ of length $y$. Regarding the choice of challenges, we note that to ensure the independence of output bits we need to avoid repetitions of challenges. A simple increment function allows us to easily expand multiple challenges from an initial challenge, while avoiding collisions and covering the whole space of possible challenges optimally. Under the assumption that the number of challenges to expand is implicitly known for each reconstruction of a response, this makes it possible to only store the starting challenge and derive all following challenges.

Thus, to enable reconstruction of fixed size stable responses, we need to store only the starting challenge of size $x$ and the helper data $W$. Once all possible challenges are exhausted the PUF should not be reused[3].

Requirement 3 of the Strong PUF definition, as well as the attributes of the fuzzy extractor, must ensure that even just a one bit difference between challenges guarantees a fully independent response.

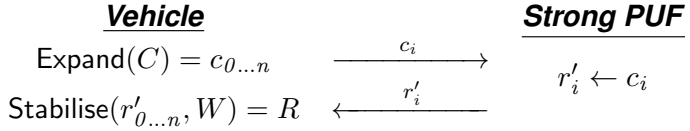



Figure 7.8: Regeneration of responses is analogous to the initial provisioning, except the previously generated helper data ($W$) is now utilized by the Stabilise() function to stabilize the response.

The details of the ECDSA key pair generation process are specified in [106]. For example a fixed amount of 320 random bits are required to deterministically build a key pair of 256 bits. Thus, we assume a need of $z = 320$ bits of stable entropy from the PUF to be able to generate a 256 bit ECDSA key pair.

Once the vehicle has constructed its key pair as outlined above, it can then build and submit a certificate signing request (the public key) to the CA through a authenticated and integrity protected channel to trigger the certification process. The CA subsequently returns a signed certificate, which completes the provisioning process of a new pseudonym.

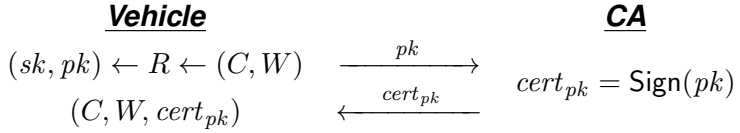



Figure 7.9: The vehicle generates an asymmetric key pair from a challenge $C$ and helper data $W$. The CA creates a certificate for the public key $pk$, which is stored in the vehicle with $C$ and $W$.

This method of secure key generation and key reconstruction from PUFs completely avoids any need for classic secure storage. The starting challenge and helper data can be stored in regular storage space. The security of the key material is fully guaranteed by the need to have access to the related PUF device with its intrinsic tamper resistant attributes.

#### 7.1.4.2 Weak PUF-based key derivation

A Weak PUF deviates from the definition of Strong PUF by allowing just one fixed CRP per PUF. It can be considered as a PUF that has a fixed built-in challenge and whenever

[3] Reconfigurable PUFs have been proposed as a desirable extension [114]

queried provides the same response. This leads to the violation of requirements 2 and 3 of the Strong PUF definition as described above. Nevertheless, even if the Weak PUF has a capacity of one single CRP, this CPR will have a useful amount of entropy. Assuming that the size of the response provides sufficient entropy for a master secret as described in Section 7.1.3.3, we can apply the same technique here.

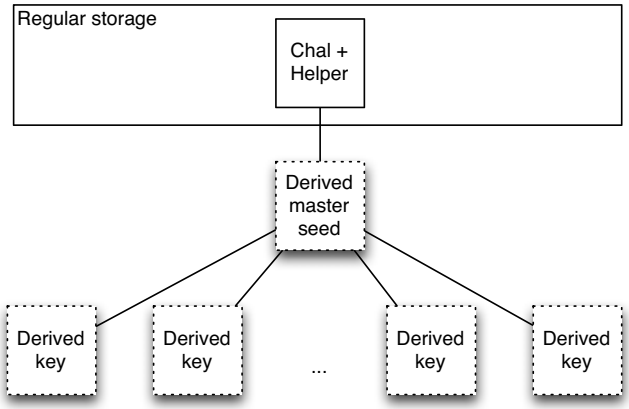



Figure 7.10: A master key gets reconstructed securely from a weak PUF using regularly stored challenges and helper data and is then used to regenerate derived keys.

An illustration of the two stage key derivation process is shown in Figure 7.10. First, a master key is derived from the response of a Weak PUF. Then, this master key is used as a seed to derive the key material for multiple pseudonyms. For instance, the PUF response could be used as the “input keying material” for the *Extract* function of HKDF [111].

### 7.1.5 Discussion

The presented solutions for the secure storage of key material for pseudonyms employ Key Derivation Functions (KDF) and Physical Unclonable Functions (PUF) to achieve multiple levels of efficiency improvements. The two major aspects for the evaluation of the solutions are the *storage requirements* and the *security properties* with respect to attackers with different capabilities.

In Table 7.1 we summarize the storage requirements of the proposed solutions for the secure storage of $k$ keys. Based on the assumption of storing $k = 105120$ keys, the baseline classic secure storage scenario would require approximately 3.2 Mbytes of secure storage space. An encrypted data structure, as described in Section 7.1.3.2, would allow to drastically reduce the amount of secure storage. In this scenario, the full 3.2 Mbytes of encrypted key material still has to be stored, but it can be stored in regular memory.
The use of a key derivation function removes this requirement of regular storage by relying purely on a master seed value, which is used to generate key material on-the-fly.
The solution based on the application of a Strong PUF does not require any classic secure

Table 7.1: Storage size overview for $k$ keys

| | Secure Storage | Regular Storage | Comments |
|---|---|---|---|
| Classic secure storage, Section 7.1.3.1 | $k$ private keys | – | ECDSA private key size ≈ 256 bit |
| Encrypted storage, Section 7.1.3.2 | 1 master key | $k$ encrypted private keys | Master symmetric key size ≈ 128 bit |
| Key derivation, Section 7.1.3.3 | 1 master seed | – | Master seed size ≈ 320 bit |
| Strong PUF-based secure storage, Section 7.1.4.1 | – | $k$ challenges, $k$ helper data | Size of challenge and helper data is highly dependent on PUF construction. Chen et al. recommend challenge sizes of 64 or 128 bits for a BR-PUF [115]. |
| Weak PUF-based key derivation, Section 7.1.4.2 | – | helper data | A Weak PUF does not necessarily require a challenge. Maes et al. [116] list $y = 2052$ bits of helper data for a response of 128 bits from an RO-PUF |

storage device at all. Instead, it is possible to rely solely on the intrinsic security of the PUF construction. However, the amount of regular storage space required to regenerate keys is larger than the raw amount of private keys. This is due to the need for helper data, which is required to stabilize the readings of responses from the noisy hardware constructions of PUFs. The exact amount of required helper data and the size of challenges are highly dependent on the attributes of a given PUF and also on algorithmic choices of the fuzzy extractor.
Finally, we see that a combination of PUF and KDF techniques even allows us to present a solution that technically does not require any secure or regular storage at all. The Weak PUF using just one challenge-response pair, returns its response without any explicit challenge, simply by virtue of being powered on.

The second criteria to compare the proposed solutions is the resilience against attackers with different levels of capabilities (see Section 7.1.2.3). Table 7.2 gives an overview of the security properties. We see that all solutions guarantee the basic requirement of denying any access to the key material to an attacker who has access to the regular unsecured filesystem.
As described in Section 7.1.2.3, the next level of attacker capability grants the attacker read-only access to arbitrary regions of OBU memory. The attacker might have found an exploitable bug in the software and injects malicious code to extract valuable data. We see weaknesses in three of the proposed solutions, due to the fact that these rely on a single piece of master secret to derive key material. This master secret (a master key or

Table 7.2: Key stealing protection under different attacker capabilities

| | Filesystem access | Memory access | Full control of OBU |
|---|---|---|---|
| Classic secure storage, Section 7.1.3.1 | safe | current key accessible | rate limitation possible |
| Encrypted storage, Section 7.1.3.2 | safe | all keys accessible | all keys accessible |
| Key derivation, Section 7.1.3.3 | safe | all keys accessible | all keys accessible |
| Strong PUF-based secure storage, Section 7.1.4.1 | safe | current key accessible | rate limitation possible |
| Weak PUF-based key derivation, Section 7.1.4.2 | safe | all keys accessible | all keys accessible |

a master seed) has to be extracted from a classic secure storage device or from a PUF and is identical for all keys that are derived by the system. An attacker with the capability to observe the address space of the application can potentially copy this master secret during the derivation process of any key. The attacker can then derive all possible keys based on this master secret.

Only the pure classic secure storage solution and the Strong PUF based solution are not affected by this issue, because these solutions derive all keys independently.
The final model grants the attacker full control over the host, which implies code execution privileges and direct access to the device. Generally, there is no way to protect the information against access by such a powerful attacker, because the storage device cannot see a difference between normal usage and usage by such a powerful attacker. One last option to offer a mitigation against malicious use could be a rate limitation mechanism, which limits the number of requests over time. For the use case of pseudonymous communication in vehicular communication it could be sufficient to only return one key per minute. Such a feature represents a viable security benefit, because the attacker can effectively only make use of the attacked device while it is online. The classic secure storage solution, as well as the Strong PUF-based solution, could reasonably offer such a feature.

#### 7.1.5.1 Limitations of KDFs and PUFs

In the previous section, the comparison of the security properties listed in Table 7.2 shows that the existence of a single master secret, as it is the case in the KDF-based solutions, represents a disadvantage under certain attacker models. Another issue to consider is the limitation of the number of keys to derive from one single master key. It is advisable to rekey the system after a certain amount of keys was derived. The rekeying interval depends on the construction of the underlying algorithm used in the KDF. This also highlights the abstract disadvantage of having to rely on additional cryptographic algorithms compared to the solutions that access keys without intermediary derivation steps. More

exposure to cryptographically strong algorithms naturally implies more risk of being affected by a discovery of a weakness in such algorithms.

Similar concerns are valid for PUFs, where the fuzzy extraction process is comparable to a key derivation process. The complexity of these processes might enlarge the exposure to bugs and weaknesses. Moreover, there are fundamental capacity limits (i.e. challenge-response pair space) that might impede practical deployments. Since PUFs are intrinsically bound to hardware, it might be impossible to reuse (rekey) a PUF after the capacity limit is reached. This is particularly problematic for Weak PUFs with only one or a very limited number of challenge-response pairs. Controlled PUFs and Reconfigurable PUFs [114] have been proposed as solutions for this problem, but the feasibility of such constructions is hard to evaluate. A controlled PUF would be particularly desirable in the context of secure storage for the possibility to effectively implement rate limitation in hardware.

While it is not an issue for pseudonyms storage in vehicular network, we acknowledge that the speed of accessing a PUF can be a limiting factor. The secure key reconstruction from PUFs incurs a considerable amount of computational overhead for the fuzzy extraction of responses. According to [116] the execution time is in the order of magnitude of several milliseconds for an RO-PUF design. Additionally, the challenge $C$ and helper data $W$, which need to be stored for the regeneration of a stable response, are significantly larger than the plain private key. While we propose an expansion function to avoid storing all challenges $c_n$, the helper data can easily add up to several kilobytes in order to generate stable response data [116].

Another limitation of using PUFs for key generation and key storage is that PUFs are effectively read-only devices. Therefore, it is necessary for vehicles to create key pairs locally, using the response of a PUF challenge as a controlled source of entropy.

We summarize the limitations of PUFs as follows:

1. Read-only data store
2. Limited capacity
3. Readout time
4. Faith in fuzzy extractor algorithms
5. Need to store helper data

These limitation pose restrictions on the realm of possible applications for PUF-based secure storage. PUF-based solutions are consequently not suitable as a direct universal replacement for all applications of classic secure storage. Nevertheless, when these limitations are met, the use of PUF-based solutions is a secure and efficient option to replace classic secure storage.

#### 7.1.5.2 PUF integrated within an HSM

As shown in Figure 7.2, the PUF could be inside an HSM. Then, our schemes would benefit from this secure computation environment. Indeed, an HSM commonly provides secure memory, secure storage, and secure cryptographic primitives. This solution ensures that

the key is generated and used at the same place, and never leaves the HSM. In this case, one can notice that integrating the PUF inside the HSM will prevent all the key stealing attacks listed in Table 7.2.

However, an attacker with full access could still use the HSM to perform malicious actions such as signing forged message. Moreover, the PUF limitations still hold even within an HSM. For instance, the limited capacity of the challenge space triggers the question about what would happen when a CRP space is depleted. As no Reconfigurable PUF exists yet, replacing the HSM would incur a considerable cost.

Finally, we conclude that if secure computation is assumed, then the cost benefit advantage of PUF is questionable. We note that the encrypted storage model (Section 7.1.3.2) would not suffer from any limitation of the PUF-based solutions while offering a better tradeoff between secure storage and regular storage. According to Table 7.1, *encrypted storage* needs 1 private key and $k$ cipher texts, while *PUF-based* approaches require no private key but $k$ challenges and $k$ helper data. One should notice that, in terms of size, the cipher text is significantly smaller than the set of challenges and helper data.

### 7.1.6 Conclusion and Future Work

Security and privacy of vehicular communication are mandatory to ensure a successful deployment and user acceptance of cooperative Intelligent Transportation System. The current set of V2X standards foresees the use of asymmetric cryptography, digital signatures, and certificates to authenticate users. To prevent tracking and privacy leakage, vehicles frequently switch between short-term pseudonyms to provide anonymity and unlinkability. As permanent–or even frequent–connection to the PKI to retrieve new pseudonyms cannot be guaranteed, a common solution is to store enough pseudonyms for one year or longer in secure storage. However, secure storage of large amount of key material is expensive if done in secure memory of a hardware security module.

We propose to use encryption and key derivation functions to reduce the need for secure storage. Our comparison shows that the use of these techniques are effective at reducing the requirements for secure storage at the cost of reduced protection against attackers with access to host memory. We alternatively propose to use Physical Unclonable Functions (PUFs) to eliminate the need for classic secure storage entirely. Our analysis shows that PUFs can effectively replace classic secure storage if an application can operate under the limitations of a given PUF. The use-case of secure pseudonymous communication in vehicular networks is generally compatible with these limitations.

The attractiveness of PUF-based solutions is a result of potential cost savings due to the use of PUF constructions compared to more expensive secure storage. PUFs are envisioned to be cheap enough for inclusion in mass produced RFID tags or might already exist in common hardware. This represents a considerable cost-benefit advantage. Once the availability of hardware implementations increases, we expect PUF-based solutions, such as the storage solutions presented in this paper, to see widespread use in practical applications.

As future work, we point out that detailed assumptions about the behavior of PUFs are often hard to verify. In this paper we require two properties about PUFs that allow us to implement optimizations and make assumptions about the security of the overall system:

1. A one bit difference between two challenges is enough to guarantee completely independent responses. Knowledge of related (not randomly selected) challenges does not affect the unpredictability of responses.
2. Knowledge of helper data does not reveal any information about the expected response from a PUF.

These attributes are implied by the Strong PUF requirement 2 and by the fuzzy extraction algorithm goals. But usually no explicitly guarantees of these attributes are given in the design documents of concrete PUF constructions.

Applications of secure storage in vehicular OBUs often involve full Hardware Security Modules (HSM) to provide secure computation in addition to secure storage. Rate limitation and a limited lifetime of certificates do allow operation without secure computation. It remains an open question, if a PUF-based secure storage solution can be augmented to offer secure computation, while retaining a cost-benefit advantage over classic implementations in an HSMs.

Development of PUF constructions is a very active area of research and we hope that new developments might remove some of the aforementioned limitations.

## 7.2 The Impact of Security on Cooperative Awareness

### 7.2.1 Introduction

Vehicular networking has received great attention by academia, industry, and politics. It brings the promise to make our driving safer, more efficient and environment-friendly, and last but not least, also more comfortable. These goals can only be achieved if vehicular networking is based on a technology that is robust against malicious attackers, and this need was stressed very early in publications like [117].

A central aspect is authentication and integrity protection for messages. It should be ensured that only valid vehicles can send messages that other vehicles will accept as genuine, and that attackers cannot modify or tamper with sent messages. Both the IEEE 1609.2 standard and its corresponding counterpart for Europe, ETSI TS 103 097, foresee the use of digital signatures using Elliptic Curve DSA (ECDSA) and the NISTp256 curve as cryptographic basis. Furthermore, both standards foresee a public key infrastructure where Certificate Authorities (CAs) issue digital certificates for vehicles that attest the validity of vehicle's key pairs.

Vehicles own asymmetric key pairs and certificates, and use those keys to attach signatures and certificates to messages. This attachment has a direct influence on communication reliability. The size of this added security payload is 65 bytes for the signature and

140 bytes for a certificate. As [118, 119] discuss, such an increase of message size will lead to an increase of packet collisions — especially on a congested channel. Both papers suggest that it is not a clever strategy to attach a certificate to every single message.

Once a receiver *A* obtained a certificate of a neighboring vehicle *B*, further certificates attached to subsequent messages of *B* are redundant and can be omitted. However, as vehicular networks typically use broadcast communication to an unspecified set of neighboring vehicles, *A* has generally no means to know whether all receivers already know its certificate. So if *A* omits a certificate from a message, this creates the risk that a receiver not knowing the certificate cannot validate the public key of *A*, and then needs to discard the message. This creates a security-induced "cryptographic packet loss" in contrast to network-induced "network packet loss". Attaching less certificates to subsequent messages of *A* increases the cryptographic packet loss while reducing network packet loss. Attaching certificates to every single message removes cryptographic packet loss while potentially increasing network packet loss.

The problem we want to address is the search for an optimal strategy that balances this trade-off to achieve a minimum overall packet loss. Previous approaches like [118–120] have investigated different approaches for certificate omission that will be discussed in the next sections. They have, however, one significant drawback. Their evaluation is based on the number of packets that is lost, and not on the impact that this has on application performance. One notable exception is [121, 122] that looks at one specific application to investigate how many crashes different omission schemes can help to prevent.

We take a more general approach that is using so-called *awareness quality* as a metric to compare different strategies. Awareness quality, as introduced in [123] looks at the information that a vehicle has about a specific driving situation based on the messages it received. It compares the known positions of vehicles as reported in received messages with the ground truth, i.e., the real positions of vehicles. The more this deviates (e.g., because of lost messages), the more inaccurate reactions of various applications that rely on cooperative awareness among vehicles will be.

By using this metric we are able to predict which certificate omission strategy works best for a variety of applications.

### 7.2.2 Awareness Quality

Periodically exchanging Cooperative Awareness Messages (CAM)[4] establishes up-to-date awareness of all surrounding vehicles and their status. Awareness in the road traffic context refers to the relation between knowledge of vehicles that *are* stored in a vehicle's neighbor table and the knowledge of vehicles that *should be* stored. The corresponding awareness requirements depend on the active safety applications [124], and of course on the network load, i.e. the acceptable awareness varies significantly under low and high load. Under low load, the awareness should be equal or close to 100% within the

[4] Also known as Basic Safety Message (BSM) or one-hop safety beacon.

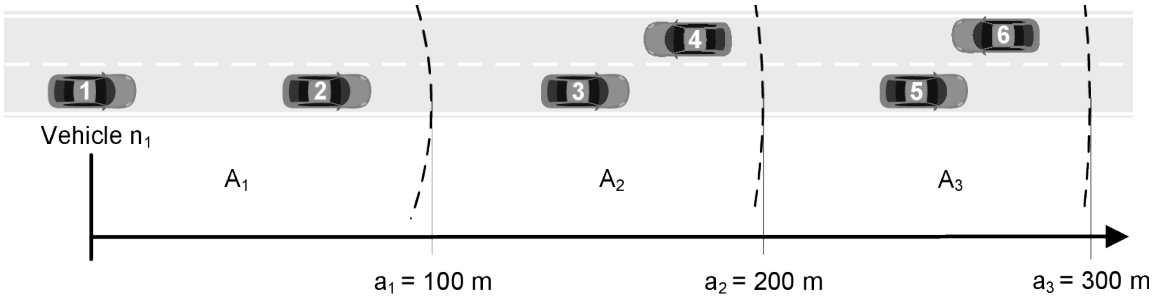



Figure 7.11: Example for the awareness quality from the viewpoint of vehicle $n_1$.

transmission range. Under high load, there should be a suitable trade-off, where the limited communication resources should be focussed to achieve high awareness in the most safety-critical area(s). For example, on highways, the vehicles travelling in the same direction should be known. In urban areas, especially at intersections, the cross-traffic should be known since they pose the highest risks there. To quantify the achievement of these different requirements, a more fine-grained awareness metric is defined in the following. To compute the Awareness Quality (AQL) each vehicle reports the level of awareness as the fraction of known vehicles based on a given validity. This validity may increase with the distance to the concerned vehicles. For the sake of simplicity, this can be assumed to increase linearly.

#### 7.2.2.1 Notation

First of all, areas of different awareness requirements around receiving vehicles are defined. Simplified, these areas are rings. The most safety-relevant ones are the rings between $0$ and $100$ m, $100$ and $200$ m, and $200$ and $300$ m, as shown in Figure 7.11. Safety areas $A_k$ are assumed to be circular with a size of

$$A_k = \pi * (a_k^2 - a_{k-1}^2), k \in \mathbb{N}$$

with $k$ denoting the identifier of the area, which are assumed to have equidistant radiuses, i.e.

$$a_k = a_1 * k, k \in \mathbb{N}$$

with the initial radius $a_1$ being $100$ m in the previously described example.

At time $T$ and for a certain vehicle $i$, the awareness within area $k$ is defined as

$$Awareness_k^T(i) = \frac{|\mathcal{N}_k^T(i) \cap \mathcal{V}_k^T(i)|}{|\mathcal{V}_k^T(i)|} \tag{7.1}$$

with $\mathcal{V}_k^T(i)$ denoting the set of vehicles being within area $k$ (i.e. the ground truth) and $\mathcal{N}_k^T(i)$ denoting the set of discovered neighbors within area $k$. Note that the intersection $\mathcal{N}_k^T(i) \cap \mathcal{V}_k^T(i)$ is required to eliminate the vehicles that are still in the neighbor table but have actually moved out of the respective area. This ensures that the fraction is always less or equal to one.

To establish the set $\mathcal{N}_k^T(i)$, we define the following constraints. Vehicle $j$ is a neighbor of $i$ within area $k$ at time $T$, with $d_{ij} = distance(i, j)$ and the $k$-th safety area:

$$j \in \mathcal{N}_k^T(i) \Leftrightarrow a_{k-1} \leq d_{ij} < a_k$$

In order to measure the Awareness Quality over time, the awareness is summed up over all vehicles and divided by the number of all vehicles for all time steps $t \in T$, i.e.

$$AQL(T, k) = \frac{\sum_{j=1}^{T} \sum_{i \in \mathcal{V}} Awareness_k^T(i)}{|T| \times |\mathcal{V}|} \tag{7.2}$$

For $AQL(T, k)$, the number of probes in the nominator is exactly the same as in the denominator. As these probes $Awareness_k^T(i) \in [0; 1]$, the resulting value of the AQL is also in the interval $[0; 1]$.

#### 7.2.2.2 Remarks

There are various reasons why the $Awareness$ can be less than one. For example, a low penetration rate degrades this ratio significantly. However in this paper, it is assumed that the penetration rate is close to $100\%$, otherwise high channel load may not be reached. Therefore, only communication-related issues are considered. In high load situations, packet loss occurs due to interference. The packet loss may even occur at low distances between sender and receiver which would most likely prevent active safety applications from working properly.

#### 7.2.2.3 Example

An example for the $Awareness$ for a single vehicle is depicted in Figure 7.11. Six vehicles $n_1, n_2, \ldots, n_6$ take part in this scene. Assuming $n_6$ just came into range of $n_1$ and no CAM has been received at the measurement time $T = 1$. There are three safety areas $A_1, A_2, A_3$ defined, equidistantly separated by $a_1 = 100$ m.

$$(Awareness_1^1, \ldots, Awareness_3^1) = (1, 1, 0.5)$$

since

$$\left(|\mathcal{N}_1^1(n_1)|, \ldots, |\mathcal{N}_3^1(n_1)|\right) = (1, 2, 1)\,,$$
$$\left(|\mathcal{V}_k^1(n_1)|, \ldots, |\mathcal{V}_3^1(n_1)|\right) = (1, 2, 2)\,.$$

Note that the reason for not knowing vehicle $n_6$ could be also due to a packet collision or increased signal attenuation by the vehicles in-between.

For more explanation on field trial implementations of this metric, the reader is referred to [125].

#### 7.2.2.4 Impact of Certificate Omission

Originally, the AQL had been established aiming at the comparison of approaches to improve channel usage. Thus, effects of packet loss, lower transmission power or higher message intervals on the set of neighbors ($\mathcal{N}_k^T(i)$) have been investigated. The constraints whether to know a neighbor or not only depends on whether the information has been received or not. Security considerations like presence of signatures and/or certificates have not been taken into consideration, which requires a separate analysis. So, in the following section, the impact of certificate omission will be discussed.

### 7.2.3 Certificate Omission Schemes

We conduct a simulation study to evaluate the use of Awareness Quality (AQL) for the assessment of the following certificate omission schemes:

- No omission of certificates (NoOm): This scheme serves as a baseline as it performs no omission.
- Periodic omission of certificates (POoC) [119]: The idea of the POoC is to add the certificate every $n$ beacons.[5] We evaluate two certificate periods of 3 seconds and 10 seconds.
- Neighbor-based certificate omission (NbCO) [118]: This scheme considers the context of a vehicle in the omission decision. The idea of NbCO is to only attach the certificate to beacons if there is a change in the neighbor table.
- Congestion-based certificate omission (CbCO) [120]: This scheme considers the load of the communication channel as the guiding metric. If the communication channel is free, there is no need to omit certificates to reduce the load on the channel. And if the communication channel is congested, then the communication load is reduced by aggressively omitting certificates. We evaluate two functions, which are used to adapt the omission frequency based on an implicit channel model and the neighbor table of vehicles: quadratic and linear.

[5] called *certificate period* in the original paper

Table 7.3: Simulation parameters

| Parameter | Value |
|---|---|
| Field size | 3 km $\times$ 3 km |
| MAC | 802.11p, 6 MBit/s |
| Fading | Rayleigh |
| Pathloss | Two-ray ground |
| Noise | Additive |
| Simulation time | 60 s |
| Transmit power | 20 dB |
| Beaconing frequency | 10 Hz |
| Payload Size | 50 Bytes |
| Number of nodes | 100, . . . , 1300 |
| Node placement | STRAW [127], |
| Node mobility | STRAW [127], |

### 7.2.4 Simulation Setup

The simulation is based on the JiST/SWANS [126] software with extensions by Ulm University.[6] The simulation environment provides 802.11p radio simulation and a realistic vehicular mobility model called STRAW [127], which uses map data from the U.S. Census Bureau. This simulation package allows us to efficiently simulate scenarios with a high density of vehicles [128]. We use a $3$ km by $3$ km urban city map in Suffolk County, U.S.A., which is the same scenario as used in previous research in omission schemes [118, 120].

In our simulation we consider only the transfer of one-hop beacon messages over one radio channel. While one-hop beacon messages will not be the only safety messages, we assume that these messages will dominate the load. The configuration of the 802.11p communication channel is set to $6$ MBit/s with a fixed transmission power of $20$ dB. The basic parameters for our simulation are in line with previous works by Schoch et.al [118] and IEEE 1609.2 [56]. A summary of relevant parameters is given in Table 7.3. For the format of beacon messages we follow the Basic Safety Message (BSM) format as specified in SAE J2735 [129]. We do not consider any optional Part II attributes of the BSM format or optional parts of the 1609.2 message format. The security services specified in IEEE 1609.2 offer different options for the cryptographic additions to messages. From these options we selected the compressed representation of NIST P-256 keys and signatures. We do not consider certificate chains in this study, but we note that certificates chains would increase the benefit of certificate omission as the crypto payload would get even larger. A summarized description of the cryptographic additions to our simulated messages is included in Table 7.4. Adding the $45$ bytes BSM and $5$ bytes for headers in the payload to the cryptographic material, the total size of one beacon message is $255$ bytes with certificate and $115$ bytes if the certificate is omitted.

[6] Website: http://www.vanet.info

Table 7.4: Cryptographic settings

| Parameter | Value |
|---|---|
| PKAlgorithm | NIST P-256 |
| ECC Key Type | compressed |
| Cert Size | 140 Bytes |
| Signature Size | 65 Bytes |

Beacons are sent every $100$ milliseconds, as suggested by SAE J2735 [129]. The lifetime of beacon messages in the neighbor table of vehicles is fixed at $150$ milliseconds. If the signature of a beacon cannot be verified, i.e. due to a missing certificate, the beacon is discarded. We refer to packets lost due to unverifiability as cryptographic packet loss (CPL) [120]. The sample rate for the collection of AQL measurements is fixed at $1$ beacon cycle period, which is $100$ milliseconds in our scenario. To test the efficiency of omission schemes under high loads, we scale the number of vehicles in the simulation scenario between $100$ and $1500$ vehicles on a $3$ km x $3$ km mixed road network. We further specify two load scenarios:

- Low density: $300$ vehicles, $33$ vehicles/km$^2$
- High density: $1500$ vehicles, $166$ vehicles/km$^2$

A full simulation run executes $60$ seconds of simulation time. During this time we do not simulate pseudonym changes. We expect the rate of uncoordinated pseudonym changes to be low enough to not be a relevant factor for the bandwidth optimization of beaconing services. Coordinated protocols for pseudonym changes, e.g. MixZones [130], might exhibit similar conditions as the initial seconds of our simulations.

### 7.2.5 Average AQL Measurements

We start by investigating awareness quality (AQL) measurement over multiple rings of safety areas. As described in Section 7.2.2 and illustrated in Figure 7.11, we specify ring shaped areas around vehicles in segments of $100$ m width [125]. The AQL measurement are calculated as the average over all vehicles in the scenario and over the whole $60$ seconds of simulated time of the scenario. We collect measurements for NbCO, two variants of POoC, using a period of $3$ and $10$ for the omission scheme, and two variants of CbCO, using a linear and a quadratic adaption function. As a baseline we also measure the AQL for the NoOm scheme, which performs no omission.

In the low density scenario (Figure 7.12) the AQL starts out at almost $100$% in the safety critical area up to $100$ m around the vehicle. The AQL then gradually decreases with the distance of the rings from the vehicle until signal propagation attributes cause severe drops in the rings between $600$ m and $900$ m distance. AQL finally converges to $0$% around the $1000$ m ring. The baseline NoOm scheme stands out in this scenario as the only

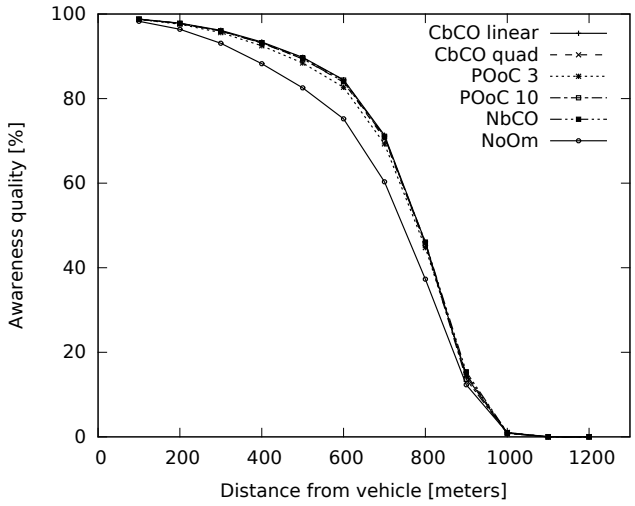



Figure 7.12: Average AQL in areas of $100$ m width around vehicles in the low density scenario

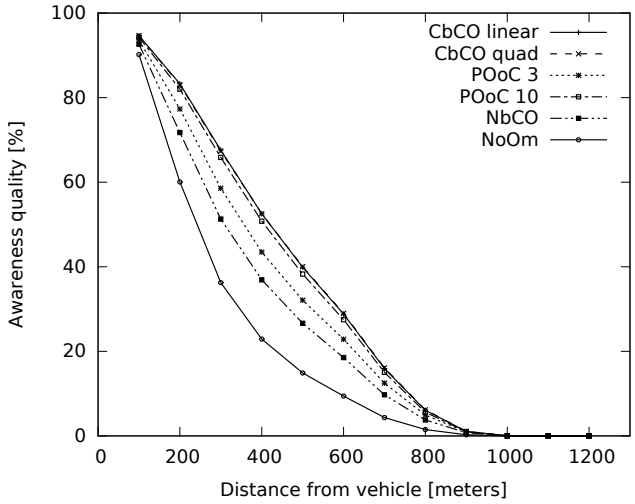



Figure 7.13: Average AQL in areas of $100$ m width around vehicles in the low density scenario

scheme with a reduced AQL compared to all omission schemes. This illustrates that even in this scenario with low density of vehicles we do see the negative effect of increased packet collisions due to consistently larger beacons.

The same scenario in a high density configuration (Figure 7.13) shows this effect more visibly. The increase of packet loss decreases the AQL very quickly. None of the schemes manages to achieve an overall AQL above 70% in the safety relevant ring up to a distance of $300$ m. At the same time it is clearly visible that the various omission schemes show different scaling behavior in such a scenario. The NbCO and POoC3 schemes in particular show worse performance than all other tested omission schemes.

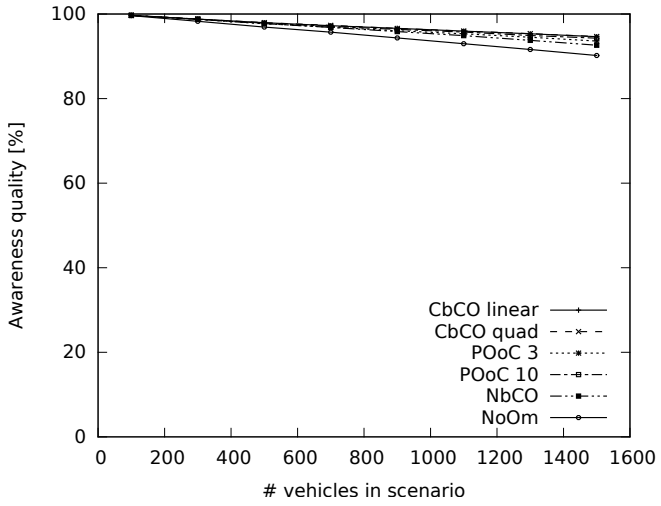



Figure 7.14: Average AQL for a safety area of $0$ m to $100$ m around vehicles under varying numbers of vehicles

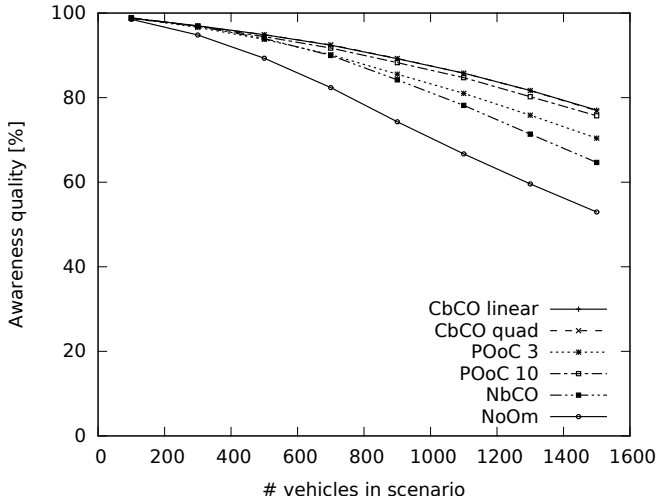



Figure 7.15: Average AQL for a safety area of $0$ m to $300$ m around vehicles under varying numbers of vehicles

Overall, these measurements match expected results from previous investigations of AQL [131] and certificate omission schemes [132]. To investigate the scalability problem in further detail we simulate the AQL in function of the number of vehicles in the scenario. Figure 7.14 shows the corresponding graphs for all schemes in the safety critical ring of $0$ m to $100$ m around vehicles. The AQL measurements show very robust performance for all schemes in this area, which confirms the absence of regressions in all these schemes with respect to proper operation in this most critical area.

In Figure 7.15 the same scenario for the ring from $200$ m to $300$ m shows more diversified results. The region up to a distance of $300$ m around a vehicle is not considered to be

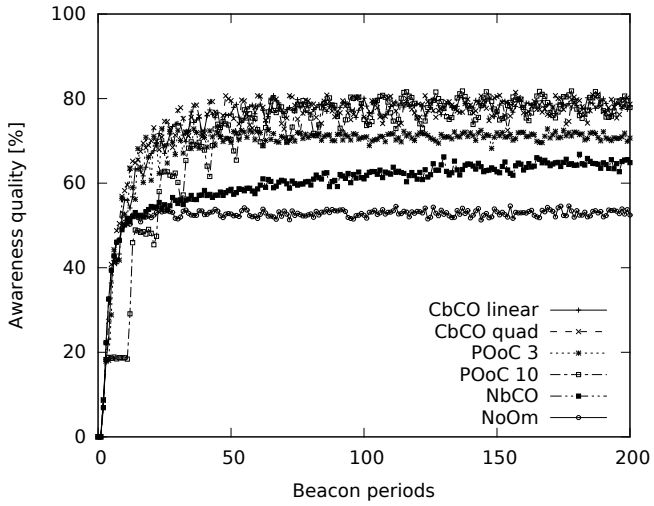



Figure 7.16: AQL measurement during the first 200 beacon periods of a high load simulation at a sampling rate of 1 per beacon cycle

safety critical but still safety relevant [125]. Like in Figure 7.13, we see decreasing performance for the POoC3, NbCO and NoOm schemes under higher load in Figure 7.15.

### 7.2.6 Time Series of AQL Measurements

The results presented so far in Figures 7.12 - 7.15 illustrate a scalability problem of not using a certificate omission and of degraded performance of some certificate omission schemes in scenarios with high densities of vehicles. But based on these measurements it is hard to identify the cause of these differences. In the preceding Figures we only see highly averaged AQL measurements, which got calculated as a combination of samples collected over all vehicles in the scenario and over the whole $60$ seconds of simulated time of the scenario. Previous research around certificate omission based on packet loss statistics faced similar problems. Even if the window of time that is used for the generation of measurements is small, the fundamental problem remains that we work with aggregated data.

The use of AQL as a metric enables us to avoid aggregation of measurements over time. Based on the sample rate of the AQL measurements it is possible to present AQL values as a time series of measurements. This is possible because AQL can report exact awareness quality values at any given point in time, something that is not possible for packet delivery statistics.

Figure 7.16 shows time series of AQL measurements, which are computed as an average over all vehicles in the scenario. Since no averaging takes place in the time domain we can see meaningful results at any given time in the simulation, even at very early stages. For better readability of the analysis we only show the initial $200$ beacon periods of the simulation in Figure 7.16. A beacon period in our simulation scenario is specified as a

fixed period of $100$ ms. A beacon period of $200$ represents $20$ seconds of simulated time. We choose the AQL sampling rate to match the beacon period. The scenario uses the high density configuration of $1500$ vehicles and we use a ring of $300$ m width from $0$ m to $300$ m distance around the vehicle in order to cover the entire security relevant area around the vehicle.

The baseline performance is again represented by the NoOm scheme. We reiterate again that not using any omission at all is clearly detrimental to the overall performance of a secure beaconing service under high load. We notice that the NbCO scheme initially performs almost identical to the NoOm scheme. This suggests that it does indeed operate almost identically as the NoOm scheme. This behavior is explicable through the high amount of unknown neighbors in the early stages of the simulation. The existence of unknown neighbors in NbCO block omissions of certificates, which increases the load on the channel and the number of packet collisions. The NbCO scheme fails to reduce the load on the channel at a time when it would be most important to back off. Consequently it takes a comparatively long time for NbCO to escape from the default behavior of NoOm.

The POoC3 scheme, which we previously identified as the third scheme with significantly degraded scalability under high load, shows performance characteristics that are independent of the behavior of NoOm and NbCO. The initial reaction time of POoC3 is competitive with other schemes, but POoC3 quickly stabilizes around an AQL level of approximately 70%. This indicates that this fixed omission works well during the initial pressure of exchanging many certificates, but during later stages it is obvious that this non-adaptive strategy leaves room for better scalability. A very notable difference can be seen between POoC3 and POoC10. The latter scheme shows very good overall scalability, matching the CbCO schemes. However in the early stages of the simulation we can identify obvious problems in the reaction time of this scheme. The period of 9 omissions for every inclusion of a certificate is clearly visible in this figure. The AQL is clearly impacted by this long period of omissions. Nevertheless, once the vehicles know the certificates of nearby vehicles, the scalability of POoC10 is on par with the CbCO schemes, which use the available bandwidth optimally among the tested schemes.

To understand the impact of verifiable packets, in particular with respect to the POoC schemes, it is useful to look at the same results without discarding unverifiable packets. Figure 7.17 shows the resulting graph for such an analysis. In this figure we also show an omission scheme that omits all certificates, called AllOm. Under normal circumstances this would of course lead to an AQL of 0%, because certificates never get exchanged between vehicles. But it is useful to see this scheme here as an upper bound.

We see NoOm and NbCO almost unaffected by cryptographic packet loss (CPL), which indicates that these schemes are dominated by regular network packet loss (NPL). In fact we can see that the scalability attributes of all schemes eventually get dominated by NPL effects, as the AQL measurements converge to the same values as in Figure 7.16. But the reaction time at the beginning of the simulation shows critical differences. We see that the “stairs effect” of POoC10 as seen previously is eliminated if we do not consider CPL. Instead, we see almost optimal behavior approaching that of AllOm, with the exception of occasionally dropping down due to the inclusion of certificates in every 10th beacon cycle. The CbCO schemes notably perform very similar to POoC10 in this Figure, indicating that

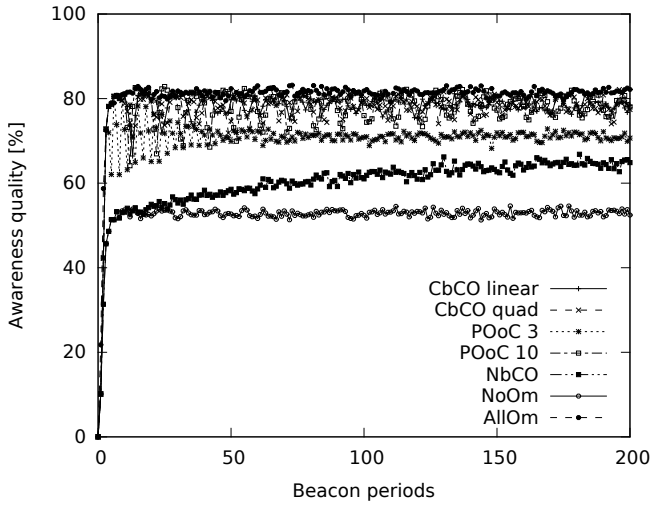



Figure 7.17: AQL measurement during the first 200 beacon periods of a high load simulation at a sampling rate of 1 per beacon cycle, not considering unverifiable packets as lost packets

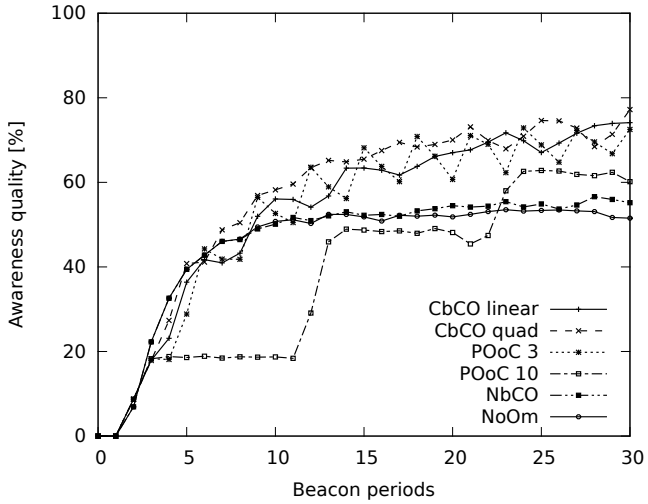



Figure 7.18: AQL measurement during the first 30 beacon periods of a high load simulation at a sampling rate of 1 per beacon cycle

the pure NPL tradeoff is as effective as POoC10. This is intuitively clear, since the CbCO schemes are algorithmically limited to behavior like POoC10 under high load [120].

### 7.2.7 Optimal Certificate Omission Scheme

The ability to zoom in on the early stages of the simulation and to see exact quality measurements at the selected sampling rate allowed us to derive a much better understanding

of the behavior of the schemes. But so far we could not see useful information about the behavior of the two CbCO schemes, beyond the observation that the schemes perform very well. In Figure 7.18 we zoom in even further by reducing the observation window to the first $30$ beacon periods, which is equivalent to $3$ seconds of simulation time at our selected beaconing interval of $100$ ms. Again we notice the "stairs effect" of POoC10 and the very similar behavior of the NoOm and NbCO schemes. Interestingly the POoC3 scheme performs very well at this early stage of the scenario, indicating that it strikes a good balance between reducing load on the communication channel and disseminating certificates to reduce CPL. We can also clearly see the oscillation of POoC3 on a period of $3$.

The two CbCO schemes seem to exhibit a similar oscillation pattern as POoC3 at this stage of the simulation. The CbCO schemes do not use a fixed omission though. The observed behavior can be explained by the fact that the CbCO schemes are adaptive POoC schemes. The schemes start out with empty neighbor tables, indicating that each vehicle is free to include certificates in beacons, since the channel is assumed to be free. With the exception of POoC10 all schemes perform similarly up to beacon period 6. Ignoring POoC10, the performance is tightly bounded by the NoOm and POoC3 schemes, indicating that we see very few omissions at this point. After the 6 beacon period mark we see a split into two groups. While NoOm and NbCO remain stagnant, we see the CbCO schemes perform similarly well as the fixed POoC3 scheme, indicating that these schemes continue to act similarly. The explanation for this can be found in the neighbor tables that slowly build up in the vehicles and gradually adjust the omission period to higher levels. Neighboring vehicles that send unverifiable beacons are not added to the neighbor table, leading the CbCO schemes to keep the omission rate at a low value. With this behavior the CbCO schemes apparently strike the best balance between NPL and CPL. Starting out with no omission and gradually increasing the omission rate only if two conditions are met:

- There are many neighbor vehicles around the vehicle, implying that the communication channel is congested,
- The neighbors send verifiable beacons, implying that the vehicles know each other and certificate omission will not have a negative effect.

A remaining uncertainty is the competitive behavior of the two CbCO scheme amongst each other. To find an answer to this question we isolate the two graphs for linear CbCO and quadratic CbCO in Figure 7.19. We know that these two schemes converge to very similar scalability properties overall. In terms of reaction time we can however identify small differences. The use of a more aggressive quadratic adaption function leads to better reaction times in a situation with many new neighbors appearing around a vehicle. In this Figure we also show error bars for the AQL measurement to illustrate the spread of AQL over the vehicles in the scenario. Since our simulation was configured with a beacon lifetime of less than two beacon periods we see every single lost beacon as a degradation of the AQL. The high standard deviation of up to 20% is the result.

Finally, with the availability of AQL as a fine-grained and exact way to investigate edge-cases of certificate omission, we show another edge case that indicates CbCO quad to

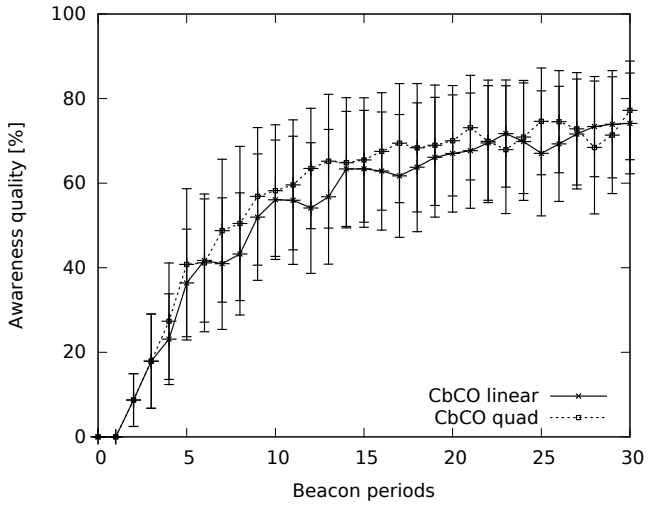



Figure 7.19: Comparative AQL measurement of CbCO linear and CbCO quad during the first 30 beacon periods of a high load simulation at a sampling rate of 1 per beacon cycle

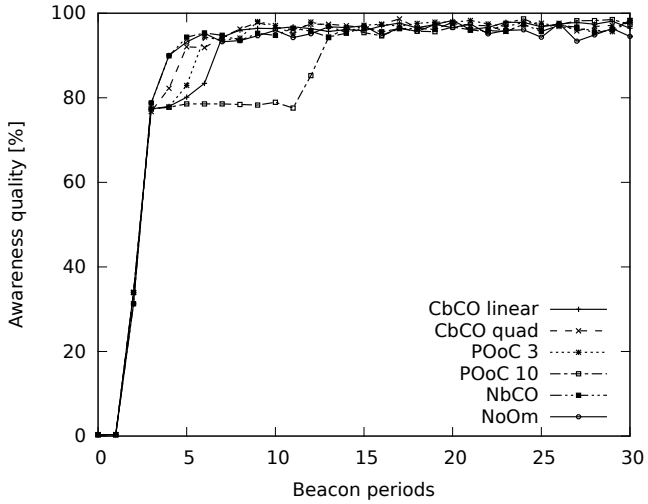



Figure 7.20: AQL measurement during the first 30 beacon periods of a low load simulation at a sampling rate of 1 per beacon cycle

be superior to CbCo linear. In Figure 7.20 we show the first $30$ beacon periods of a scenario with a low density of cars. It is expected that all schemes perform very well in this scenario, with Figure 7.14 suggesting that we should see AQL values around $98\%$. But we can see that some schemes struggle in the first couple of beacon periods. The NoOm and NbCO perform best in this scenario, followed by CbCO quad, POoC3, CbCO linear and trailed with a large margin by POoC10. The performance of CbCO quad clearly beats the performance of CbCO linear in this edge case.

Table 7.5: Performance of Omission Schemes

| Name | Reactivity | Scalability |
|---|---|---|
| No omissions (NoOm) | + + | - - |
| Neighbor-based (NbCO) | + + | - |
| Periodic Omission, $\alpha = 3$ (POoC-3) | + | - |
| Periodic Omission, $\alpha = 10$ (POoC-10) | - - | + + |
| Congestion-based, linear adaption (CbCO-linear) | + | + + |
| Congestion-based, quadratic adaption (CbCO-quad) | + + | + + |

We conclude that, thanks to AQL measurements we could identify quadratic CbCO as the preferred choice in high load scenarios and as a near optimal choice for low load scenarios. Our results are summarized in Table 7.5.

### 7.2.8 Conclusions

Security of vehicular networking is mandatory to provide robust safety applications. It is important to investigate the impact of security mechanisms on safety applications. In this paper, we analyzed the impact of certificate omission mechanisms from an application-level perspective. We used Awareness Quality to compare five omission schemes, and concluded that the Congestion-based certificate omission scheme with quadratic adaption function is the best-suited for safety applications. It provides the optimal combination of awareness quality, scalability and reactivity.

This work also demonstrated that Awareness Quality allows precise measurements of the scenarios' state at any given time. This helps to expose the intrinsic behavior of the studied schemes. We encourage security researchers to apply this metric to assess the impact of security mechanisms on cooperative safety applications. We also hope to see complementary work on top-down approaches that consider application requirements during protocol design.

# Bibliography

[1] ETSI TC ITS, "ETSI TS 102 731 v1.1.1 - intelligent transport systems (ITS); security; security services and architecture," Standard, TC ITS, 2010. [Online]. Available: http://www.etsi.org/deliver/etsi_ts/102700_102799/102731/01.01.01_60/ts_102731v010101p.pdf

[2] "Car2car communication consortium manifesto," http://www.car-2-car.org/.

[3] ETSI TR 102 638, "Intelligent transport systems (ITS); vehicular communications; basic set of applications; definitions," Jun. 2009.

[4] R. van der Heijden, S. Dietzel, and F. Kargl, "Misbehavior Detection in Vehicular Ad-hoc Networks," in *1st GI/ITG KuVS Fachgespräch Inter-Vehicle Communication (FG-IVC 2013)*, Innsbruck, Austria, February 2013.

[5] J. Hortelano, J. Ruiz, and P. Manzoni, "Evaluating the usefulness of watchdogs for intrusion detection in vanets," in *Communications Workshops (ICC), 2010 IEEE International Conference on*, may 2010, pp. 1 –5.

[6] S. Marti, T. J. Giuli, K. Lai, and M. Baker, "Mitigating routing misbehavior in mobile ad hoc networks," in *Proceedings of the 6th annual international conference on Mobile computing and networking*, ser. MobiCom '00. New York, NY, USA: ACM, 2000, pp. 255–265. [Online]. Available: http://doi.acm.org/10.1145/345910.345955

[7] A. Hamieh, J. Ben-Othman, and L. Mokdad, "Detection of radio interference attacks in VANET," in *IEEE Global Telecommunications Conference, 2009. GLOBECOM 2009.* IEEE, 30 2009-dec. 4 2009, pp. 1 –5.

[8] H.-c. Hsiao, A. Studer, C. Chen, A. Perrig, F. Bai, B. Bellur, and A. Iyer, "Flooding-resilient broadcast authentication for VANETs," in *Proceedings of the 17th annual international conference on Mobile computing and networking (MobiCom '11)*. Las Vegas, Nevada, USA: ACM Press, 2011, p. 193. [Online]. Available: http://dl.acm.org/citation.cfm?doid=2030613.2030635

[9] M. Raya, P. Papadimitratos, V. D. Gligor, and J.-P. J.-P. Hubaux, "On Data-Centric Trust Establishment in Ephemeral Ad Hoc Networks," in *2008 IEEE INFOCOM - The 27th Conference on Computer Communications*. Ieee, Apr. 2008, pp. 1238–1246.

[10] G. Shafer, *A mathematical theory of evidence*. Princeton university press Princeton, 1976.

[11] A. P. Dempster, "Upper and lower probabilities induced by a multivalued mapping," *The annals of mathematical statistics*, vol. 38, no. 2, pp. 325–339, 1976.

[12] M. Raya, P. Papadimitratos, I. Aad, D. Jungels, and J.-p. Hubaux, "Eviction of Misbehaving and Faulty Nodes in Vehicular Networks," *IEEE Journal on Selected Areas in Communications*, vol. 25, no. 8, pp. 1557–1568, Oct. 2007.

[13] G. Guette and B. Ducourthial, "On the Sybil attack detection in VANET," in *2007 IEEE Internatonal Conference on Mobile Adhoc and Sensor Systems*. IEEE, Oct. 2007, pp. 1–6. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=4428742

[14] N.-W. Lo and H.-C. Tsai, "Illusion Attack on VANET Applications - A Message Plausibility Problem," in *2007 IEEE Globecom Workshops*. Washington, DC: IEEE, Nov. 2007, pp. 1–8. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=4437823

[15] H. Stübing, A. Jaeger, N. Bißmeyer, C. Schmidt, and S. A. Huss, "Verifying mobility data under privacy considerations in Car-to-X communication," in *Proceedings of 17th ITS World Congress*, 2010.

[16] A. Jaeger, N. Bißmeyer, H. Stübing, and S. a. Huss, "A Novel Framework for Efficient Mobility Data Verification in Vehicular Ad-hoc Networks," *International Journal of Intelligent Transportation Systems Research*, vol. 10, no. 1, pp. 11–21, Aug. 2011. [Online]. Available: http://www.springerlink.com/index/10.1007/s13177-011-0038-9

[17] P. Golle, D. Greene, and J. Staddon, "Detecting and correcting malicious data in vanets," in *Proceedings of the 1st ACM international workshop on Vehicular ad hoc networks*, ser. VANET '04. New York, NY, USA: ACM, 2004, pp. 29–37. [Online]. Available: http://doi.acm.org/10.1145/1023875.1023881

[18] T. Leinmüller, C. Maihöfer, E. Schoch, and F. Kargl, "Improved security in geographic ad hoc routing through autonomous position verification," in *Proceedings of the 3rd international workshop on Vehicular ad hoc networks - VANET '06*. New York, New York, USA: ACM Press, 2006, pp. 57–66. [Online]. Available: http://portal.acm.org/citation.cfm?doid=1161064.1161075

[19] T. Leinmüller, E. Schoch, F. Kargl, and C. Maihöfer, "Decentralized position verification in geographic ad hoc routing," *Security and Communication Networks*, vol. 3, no. 4, pp. 289–302, 2010. [Online]. Available: http://dx.doi.org/10.1002/sec.56

[20] S. Chang, Y. Qi, H. Zhu, J. Zhao, and X. Shen, "Footprint: Detecting Sybil Attacks in Urban Vehicular Networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 23, no. 6, pp. 1103 – 1114, 2012.

[21] J. Rezgui and S. Cherkaoui, "Detecting faulty and malicious vehicles using rule-based communications data mining," in *2011 IEEE 36th Conference on Local Computer Networks*. IEEE, Oct. 2011, pp. 827–834. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6115558

[22] N. Biß meyer, J. Njeukam, J. Petit, and K. M. Bayarou, "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*. New York, New York, USA: ACM Press, 2012, pp. 73–82.

[23] J. Grover, V. Laxmi, and M. Gaur, "Misbehavior detection based on ensemble learning in vanet," in *Advanced Computing, Networking and Security*, ser. Lecture Notes in Computer Science, P. Thilagam, A. Pais, K. Chandrasekaran, and N. Balakrishnan, Eds. Springer Berlin / Heidelberg, 2012, vol. 7135, pp. 602–611.

[24] H. Stübing, J. Firl, and S. A. Huss, "A two-stage verification process for Car-to-X mobility data based on path prediction and probabilistic maneuver recognition," in *2011 IEEE Vehicular Networking Conference (VNC)*. IEEE, Nov. 2011, pp. 17–24.

[25] M. Mahmoud and X. Shen, "Stimulating cooperation in multi-hop wireless networks using cheating detection system," in *INFOCOM, 2010 Proceedings IEEE*, march 2010, pp. 1 –9.

[26] R. K. Schmidt, T. Leinmüller, E. Schoch, A. Held, and G. Schäfer, "Vehicle behavior analysis to enhance security in vanets," in *Proceedings of the 4th Workshop on Vehicle to Vehicle Communications (V2VCOM 2008)*, 2008.

[27] D. Antolino Rivas, J. M. Barceló-Ordinas, M. Guerrero Zapata, and J. D. Morillo-Pozo, "Security on VANETs: Privacy, misbehaving nodes, false information and secure data aggregation," *Journal of Network and Computer Applications*, vol. 34, no. 6, pp. 1942–1955, Nov. 2011. [Online]. Available: http://linkinghub.elsevier.com/retrieve/pii/S1084804511001317

[28] M. Ghosh, A. Varghese, A. Gupta, A. A. Kherani, and S. N. Muthaiah, "Detecting misbehaviors in VANET with integrated root-cause analysis," *Ad Hoc Networks*, vol. 8, no. 7, pp. 778–790, Sep. 2010. [Online]. Available: http://linkinghub.elsevier.com/retrieve/pii/S157087051000034X

[29] K. Koscher, A. Czeskis, F. Roesner, S. Patel, T. Kohno, S. Checkoway, D. McCoy, B. Kantor, D. Anderson, H. Shacham, and S. Savage, "Experimental security analysis of a modern automobile," in *Security and Privacy (SP), 2010 IEEE Symposium on*, may 2010, pp. 447 –462.

[30] ETSI - European Telecommunications Standards Institute, "Intelligent transport systems (ITS); communications architecture," ETSI, European Norm EN 302 665, September 2010.

[31] ——, "Intelligent transport systems (ITS); security; threat, vulnerability and risk analysis (TVRA)," ETSI, Technical Report TR 102 893, June 2010.

[32] T. Leinmüller, R. Schmidt, E. Schoch, A. Held, and C. Schafer, "Modeling roadside attacker behavior in VANETs," *GLOBECOM Workshops, 2008 IEEE*, pp. 1 –10, 30 2008-dec. 4 2008.

[33] ETSI - European Telecommunications Standards Institute, “Intelligent transport systems (ITS); vehicular communications; basic set of applications; definitions,” ETSI, Technical Report TR 102 638, June 2009.

[34] N. Bißmeyer, C. Stresing, and K. Bayarou, “Intrusion detection in vanets through verification of vehicle movement data,” in *Second IEEE Vehicular Networking Conference*, vol. Second IEEE Vehicular Networking Conference, December 2010.

[35] R. K. Schmidt, T. Leinmueller, E. Schoch, A. Held, and G. Schaefer, “Vehicle behavior analysis to enhance security in VANETs,” in *Proceedings of the 4th IEEE Vehicle-to-Vehicle Communications Workshop (V2VCOM2008)*, 2008.

[36] H. Stübing, A. Jaeger, N. Bißmeyer, C. Schmidt, and S. A. Huss, “Verifying mobility data under privacy considerations in Car-To-X communication,” in *ITS World Congress*, vol. 17th ITS World Congress, Busan, October 2010.

[37] PReVENT project - INTERSAFE subproject, “Requirements for intersection safety applications,” Deliverable D40.4, 2005.

[38] C. Frye, “International cooperation to prevent collisions at intersections,” *Public Roads Magazine*, vol. 65, no. 1, 2005.

[39] S. International, “Sae j2735 - dedicated short range communications (dsrc) message set dictionary,” SAE International, U.S. Department of Transportation, ITS Standards Fact Sheets J2735, September 2009.

[40] CICAS project, http://www.its.dot.gov/cicas/index.htm.

[41] B. Roessler and K. Fuerstenberg, “First European STREP on cooperative intersection safety INTERSAFE-2,” in *IEEE Intelligent Transportation Systems Conference (ITSC ’10)*, 2010, pp. 422–427.

[42] S. Lefèvre, C. Laugier, and J. Ibañez-Guzmàn, “Evaluating risk at road intersections by detecting conflicting intentions,” in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS ’12)*, 2012, pp. 4841–4846.

[43] ——, “Risk assessment at road intersections: comparing intention and expectation,” in *IEEE Intelligent Vehicles Symposium (IV ’12)*, 2012, pp. 165–171.

[44] TRACE project, “Accident causation and pre-accidental driving situations - In-depth accident causation analysis,” Deliverable D2.2, 2008.

[45] K. Vogel, “A comparison of headway and time to collision as safety indicators,” *Accident Analysis & Prevention*, vol. 35, no. 3, pp. 427–433, 2003.

[46] J. Freudiger, M. H. Manshaei, J.-P. Hubaux, and D. C. Parkes, “On non-cooperative location privacy: a game-theoretic analysis,” in *ACM Conference on Computer and Communications Security (CCS ’09)*, 2009, pp. 324–337.

[47] J. Freudiger, M. Manshaei, J.-P. Hubaux, and D. Parkes, “Non-cooperative location privacy,” *Dependable and Secure Computing, IEEE Transactions on*, vol. 10, no. 2, pp. 84–98, 2013.

[48] S. Eichler, “Strategies for pseudonym changes in vehicular ad hoc networks depending on node mobility,” in *IEEE Intelligent Vehicles Symposium (IV ’07)*, Istanbul, Turkey, june 2007, pp. 541–546.

[49] ISO, “Road vehicles – Vehicle-to-Grid Communication Interface – Part 1: General information and use-case definition (Draft),” 2012.

[50] C. Höfer, J. Petit, R. Schmidt, and F. Kargl, “Popcorn: Privacy-preserving charging for emobility,” in *Proceedings of the 2013 ACM Workshop on Security, Privacy & Dependability for Cyber Vehicles*, ser. CyCAR ’13. New York, NY, USA: ACM, 2013, pp. 37–48. [Online]. Available: http://doi.acm.org/10.1145/2517968.2517971

[51] Car 2 Car Communication Consortium, “Pilot pki: Security management message formats, version 1.0,” Tech. Rep., June 2013.

[52] A. Pfitzmann and M. Hansen, “A terminology for talking about privacy by data minimization: Anonymity, unlinkability, undetectability, unobservability, pseudonymity, and identity management,” August 2010, v0.34.

[53] N. Bissmeyer, J. Njeukam, J. Petit, and K. M. Bayarou, “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*, ser. VANET ’12. Low Wood Bay, Lake District, UK: ACM, 2012, pp. 73–82. [Online]. Available: http://doi.acm.org/10.1145/2307888.2307902

[54] N. Bißmeyer, J. P. Stotz, H. Stübing, E. Schoch, S. Götz, and B. Lonc, “A generic public key infrastructure for securing car-to-x communication,” in *18th World Congress on Intelligent Transportation Systems*. ITS America, October 2011.

[55] ETSI - European Telecommunications Standards Institute, “Intelligent transport systems (ITS); security; security services and architecture,” ETSI, Technical Standard TS 102 731, September 2010.

[56] “IEEE Draft Standard for Wireless Access in Vehicular Environments - Security Services for Applications and Management Messages,” *IEEE P1609.2/D12, January 2012*, pp. 1 – 266.

[57] A. Jaeger, N. Bißmeyer, H. Stübing, and S. A. Huss, “A novel framework for efficient mobility data verification in vehicular ad-hoc networks,” *International Journal of ITS Research, ITS Japan*, vol. 9, no. 3, September 2011.

[58] U. D. of Transportation Research and I. T. Administration, “Security credential management system design security system design for cooperative vehicle-to-vehicle crash avoidance applications using 5.9 ghz dedicated short range communications (dsrc) wireless communications,” CAMP, VSC3, www.its.dot.gov, Tech. Rep., February 2012.

[59] IEEE Computer Society, “IEEE standard specifications for public-key cryptography-amendment 1: Additional techniques,” *IEEE Std 1363a-2004 (Amendment to IEEE Std 1363-2000)*, pp. 1 –159, 2004.

[60] F. Schaub, F. Kargl, Z. Ma, and M. Weber, "V-tokens for conditional pseudonymity in VANETs," in *IEEE Wireless Communications and Networking Conference (WCNS)*, 2010.

[61] L. Fischer, A. Aijaz, C. Eckert, and D. Vogt, "Secure revocable anonymous authenticated inter-vehicle communication (SRAAC)," in *4th Conference: escar - Embedded Security in Cars*, Germany, 2006.

[62] A. Shamir, "How to share a secret," *Commun. ACM*, vol. 22, no. 11, pp. 612–613, November 1979. [Online]. Available: http://doi.acm.org/10.1145/359168.359176

[63] D. Chaum, "Blind siganture systems," US Patent 4 759 063, July, 1988.

[64] D. Jena, S. K. Jena, and B. Majhi, "A novel untraceable blind signature based on elliptic curve discrete logarithm problem," *IJCSNS*, vol. 7, no. 6, pp. 269–275, June 2007.

[65] M. Jakobsson, "Privacy vs. authenticity," PhD Thesis, University of California, San Diego, 1997.

[66] N. Alexiou, M. Laganà, S. Gisdakis, M. Khodaei, and P. Papadimitratos, "VeSPA: Vehicular Security and Privacy-preserving Architecture," in *ACM Workshop on Hot Topics on Wireless Network Security and Privacy (ACM HotWiSec)*, Budapest, Hungary, Apr. 2013.

[67] J. P. Stotz, N. Bißmeyer, F. Kargl, S. Dietzel, P. Papadimitratos, and C. Schleiffer, "PRESERVE D1.1 Security Requirements of Vehicle Security Architecture," PRESERVE consortium, Deliverable, July 2011.

[68] "Intelligent Transport Systems (ITS), Vehicular Communications (VC), Basic Set of Applications, Definitions," ETSI TR 102 638 V1.1, Tech. Rep., Jun. 2009.

[69] P. Papadimitratos, L. Buttyan, T. Holczer, E. Schoch, J. Freudiger M. Raya, Z. Ma, F. Kargl, A. Kung, and J.-P. Hubaux, "Secure Vehicular Communication Systems: Design and Architecture," *IEEE Communcations Magazine*, vol. 46, no. 11, pp. 100–109, November 2008.

[70] N. Alexiou, S. Gisdakis, M. Laganà, and P. Papadimitratos, "Towards a secure and privacy-preserving multi-service vehicular architecture," in *proocedings of the 4th International Workshop on Data Security and PrivAcy in wireless Networks (D-SPAN)*, Madrid, Spain, Jun. 2013.

[71] R. Yavatkar, D. Pendarakis, and R. Guerin, "A Framework for Policy-based Admission Control," RFC 2753 (Informational), Internet Engineering Task Force, Jan. 2000.

[72] C. Neuman, T. Yu, S. Hartman, and K. Raeburn, "The Kerberos Network Authentication Service (V5)," RFC 4120 (Proposed Standard), Internet Engineering Task Force, Jul. 2005, updated by RFCs 4537, 5021, 5896, 6111, 6112, 6113, 6649, 6806.

[73] S. Gisdakis, M. Laganà, T. Giannetsos, and P. Papadimitratos, "SEROSA: SERvice Oriented Security Architecture for Vehicular Communications," in *proceedings of the IEEE Vehicular Networking Conference (VNC)*, Dec. 2013.

[74] P. Papadimitratos, V. Gligor, and J.-P. Hubaux, "Securing Vehicular Communications - Assumptions, Requirements, and Principles," in *ESCAR 2006*.

[75] S. Cantor, J. Kemp, R. Philpott, and E. Maler, "Assertions and Protocols for the OASIS Security Assertion Markup Language (SAML) V2.0," Tech. Rep., Mar. 2005. [Online]. Available: http://docs.oasis-open.org/security/saml/v2.0/saml-core-2.0-os.pdf

[76] J. Sermersheim, "Lightweight Directory Access Protocol (LDAP): The Protocol," RFC 4511 (Proposed Standard), Internet Engineering Task Force, Jun. 2006. [Online]. Available: http://www.ietf.org/rfc/rfc4511.txt

[77] T. Moses, "XACML 2.0 Core: eXtensible Access Control Markup Language (XACML) Version 2.0," Feb. 2005.

[78] D. Boneh, X. Boyen, and H. Shacham, "Short Group Signatures," in *Crypto '04*, Santa Barbara, CA, USA, Aug. 2004.

[79] F. Schaub, F. Kargl, Z. Ma, and M. Weber, "V-tokens for Conditional Pseudonymity in VANETs," in *IEEE Wireless Communications & Networking Conference (WCNC '10)*. Sydney, Australia: IEEE, 2010. [Online]. Available: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=5506126

[80] M. Raya, P. Papadimitratos, and J.-P. Hubaux, "Securing Vehicular Communications," *IEEE Wireless Communications Magazine*, vol. 13, no. 5, pp. 8–15, 2006.

[81] S. Uppoor and M. Fiore, "Large-scale urban vehicular mobility for networking research," in *proceedings of the 3rd IEEE Vehicular Networking Conference (VNC)*, Nov. 2011.

[82] M. Raya, P. Papadimitratos, I. Aad, D. Jungels, and J.-P. Hubaux, "Eviction of misbehaving and faulty nodes in vehicular networks," *IEEE Journal on Selected Areas in Communications, Special Issue on Vehicular Networks*, vol. 25, no. 8, pp. 1557–1568, October 2007.

[83] N. Bißmeyer, J. Petit, and K. M. Bayarou, "Copra: Conditional Pseudonym Resolution Algorithm in VANETs," in *proceedings of the 10th Annual Conference on Wireless On-demand Network Systems and Services (WONS)*, Mar. 2013.

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

[85] ETSI TC ITS, "ETSI TS 102 941 v1.1.1 - intelligent transport systems (ITS); security; trust and privacy management," Standard, TC ITS, 2012. [Online]. Available: http://www.etsi.org/deliver/etsi_ts/102900_102999/102941/01.01.01_60/ts_102941v010101p.pdf

[86] ——, "ETSI TS 103 097 v1.1.1 - intelligent transport systems (ITS); security; security header and certificate formats," Standard, TC ITS, 2013. [Online]. Available: http://www.etsi.org/deliver/etsi_ts/103000_103099/103097/01.01.01_60/ts_103097v010101p.pdf

[87] B. Wiedersheim, Z. Ma, F. Kargl, and P. Papadimitratos, "Privacy in inter-vehicular networks: Why simple pseudonym change is not enough," *7th International Conference on Wireless On-demand Network Systems and Services (WONS '10)*, 2010.

[88] D. Garcia, A. Waite, R. Walsh, B. Sheppard, L. Frank, and D. Jeffers, "Certificate management entities for connected vehicle environment. public workshop read-ahead document," Research and Innovative Technology Administration, Technical report FHWA-JPO-12-038, May 2012.

[89] T. C. S. de Souza, J. E. Martina, and R. F. Custódio, "Audit and backup procedures for hardware security modules," *7th Symposium on Identity and Trust on the Internet (IDtrust '08)*, pp. 89–97, 2008. [Online]. Available: http://doi.acm.org/10.1145/1373290.1373302

[90] B. H. Kim, K. Y. Choi, J. H. Lee, and D. H. Lee, "Anonymous and traceable communication using tamper-proof device for vehicular ad hoc networks," *International Conference on Convergence Information Technology*, pp. 681–686, 2007.

[91] T. Leinmüller, L. Buttyan, J.-P. Hubaux, F. Kargl, R. Kroh, P. Papadimitratos, M. Raya, and E. Schoch, "Sevecom - secure vehicle communication," *IST Mobile and Wireless Communication Summit*, pp. 1–5, 2006.

[92] M. Wolf, A. Weimerskirch, and T. Wollinger, "State of the art: Embedding security in vehicles," *EURASIP Journal on Embedded Systems*, vol. 2007, 2007.

[93] K. Moerman, T. van Roermund, and M. Knezevic, "A realistic approach to message verification in car-to-car communication," *19th ITS World Congress*, 2012.

[94] L. Apvrille, R. El Khayari, O. Henniger, Y. Roudier, H. Schweppe, H. Seudié, B. Weyl, and M. Wolf, "Secure automotive on-board electronics network architecture," *World Automotive Congress (FISITA '10)*, May 2010.

[95] R. Pappu, "Physical One-Way Functions," Ph.D. dissertation, MIT, 2001.

[96] R. Pappu, B. Recht, J. Taylor, and N. Gershenfeld, "Physical One-Way Functions," *Science*, vol. 297, pp. 2026–2030, 2002.

[97] Y. Dodis, M. Reyzin, and A. Smith, "Fuzzy extractors: How to generate strong keys from biometrics and other noisy data," in *Advances in Cryptology – EUROCRYPT 2004*, ser. LNCS, vol. 3027, 2004, pp. 523–540.

[98] J.-P. M. G. Linnartz and P. Tuyls, "New Shielding Functions to Enhance Privacy and Prevent Misuse of Biometric Templates," *Audio-and Video-Based Biometrie Person Authentication (AVBPA '03)*, vol. 2688, pp. 393–402, 2003.

[99] R. Plaga and F. Koob, "A formal definition and a new security mechanism of physical unclonable functions," in *Measurement, Modelling, and Evaluation of Computing Systems and Dependability and Fault Tolerance*. Springer, 2012, pp. 288–301.

[100] U. Rührmair, F. Sehnke, J. Sölter, G. Dror, S. Devadas, and J. Schmidhuber, "Modeling attacks on physical unclonable functions," *17th ACM conference on Computer and communications security (CCS '10)*, pp. 237–249, 2010. [Online]. Available: http://doi.acm.org/10.1145/1866307.1866335

[101] T. Lange, "PUFFIN - the physically unclonable functions found in standard pc components project," 2013, retrieved July 10, 2013 from http://puffin.eu.org/.

[102] J. Kelsey, B. Schneier, C. Hall, and D. Wagner, "Secure applications of low-entropy keys," in *Information Security*. Springer, 1998, pp. 121–134.

[103] B. Kaliski, "RFC 2898: Pkcs# 5: Password-based cryptography specification version 2.0," *IETF, September*, 2000.

[104] N. Provos and D. Mazieres, "A future-adaptable password scheme." *USENIX Annual Technical Conference, FREENIX Track*, pp. 81–91, 1999.

[105] C. Percival, "Stronger key derivation via sequential memory-hard functions," *The Technical BSD Conference (BSDCan '09)*, May 2009.

[106] Federal Information Processing Standards, *Digital Signature Standard (DSS) - FIPS 186-3*, June 2009. [Online]. Available: http://csrc.nist.gov/publications/fips/fips186-3/fips_186-3.pdf

[107] IEEE, "IEEE standard specifications for public-key cryptography- amendment 1: Additional techniques," *IEEE Std 1363a-2004 (Amendment to IEEE Std 1363-2000)*, pp. 1–159, 2004.

[108] ISO/IEC, "Information technology - security techniques - encryption algorithms - part 2: Asymmetric ciphers," *ISO/IEC 18033-2*, 2006.

[109] B. Elaine, J. Don, and S. Miles, "SP 800-56A. recommendation for pair-wise key establishment schemes using discrete logarithm cryptography," Gaithersburg, MD, United States, Tech. Rep., 2007.

[110] H. Krawczyk, "Cryptographic extraction and key derivation: The HKDF scheme," in *Advances in Cryptology–CRYPTO 2010*. Springer, 2010, pp. 631–648.

[111] H. Krawczyk and P. Eronen, "HMAC-based Extract-and-Expand Key Derivation Function (HKDF)," Internet Requests for Comments, RFC 5869, May 2010. [Online]. Available: http://www.rfc-editor.org/rfc/rfc5869.txt

[112] L. Chen, "SP 800-56C. recommendation for key derivation through extraction-then-expansion," Gaithersburg, MD, United States, Tech. Rep., 2011.

[113] E. B. Barker and J. M. Kelsey, *Recommendation for random number generation using deterministic random bit generators (revised)*. US Department of Commerce, Technology Administration, National Institute of Standards and Technology, Computer Security Division, Information Technology Laboratory, 2007.

[114] S. Katzenbeisser, Ü. Kocabaş, V. van der Leest, A.-R. Sadeghi, G.-J. Schrijen, and C. Wachsmann, "Recyclable PUFs: Logically reconfigurable PUFs," *Journal of Cryptographic Engineering*, vol. 1, no. 3, pp. 177–186, 2011.

[115] Q. Chen, G. Csaba, P. Lugli, U. Schlichtmann, and U. Rührmair, "The bistable ring puf: A new architecture for strong physical unclonable functions," *IEEE International Symposium on Hardware-Oriented Security and Trust (HOST '11)*, pp. 134–141, 2011.

[116] R. Maes, A. Herrewege, and I. Verbauwhede, "PUFKY: A fully functional puf-based cryptographic key generator," *Cryptographic Hardware and Embedded Systems (CHES '12)*, pp. 302–319, 2012.

[117] M. Raya and J.-P. Hubaux, "The security of vehicular ad hoc networks," in *Proceedings of the 3rd ACM workshop on Security of ad hoc and sensor networks*, ser. SASN '05. New York, NY, USA: ACM, 2005, pp. 11–21.

[118] E. Schoch and F. Kargl, "On the efficiency of secure beaconing in vanets," in *Proceedings of the third ACM conference on Wireless network security*, ser. WiSec '10. New York, NY, USA: ACM, 2010, pp. 111–116.

[119] G. Calandriello, P. Papadimitratos, J.-P. Hubaux, and A. Lioy, "On the performance of secure vehicular communication systems," *Dependable and Secure Computing, IEEE Transactions on*, vol. 8, no. 6, pp. 898–912, 2011.

[120] M. P. Feiri, J. Y. Petit, and F. Kargl, "Evaluation of congestion-based certificate omission in vanets," in *Proceedings of the IEEE Vehicular Networking Conference (VNC 2012), Seoul, Korea*. USA: IEEE, November 2012, pp. 101–108.

[121] G. Calandriello, P. Papadimitratos, J.-P. Hubaux, and A. Lioy, "Efficient and robust pseudonymous authentication in vanet," in *Proceedings of the fourth ACM international workshop on Vehicular ad hoc networks*, ser. VANET '07. New York, NY, USA: ACM, 2007, pp. 19–28. [Online]. Available: http://doi.acm.org/10.1145/1287748.1287752

[122] P. Papadimitratos, G. Calandriello, J.-P. Hubaux, and A. Lioy, "Impact of vehicular communications security on transportation safety," in *INFOCOM Workshops 2008, IEEE*, 2008, pp. 1–6.

[123] R. Schmidt, R. Lasowski, T. Leinmüller, C. Linnhoff-Popien, and G. Schafer, "An approach for selective beacon forwarding to improve cooperative awareness," in *Vehicular Networking Conference (VNC), 2010 IEEE*, 2010, pp. 182–188.

[124] H. Hartenstein and K. Laberteaux, *V ANET: vehicular applications and internetworking technologies*. Wiley Online Library, 2010, vol. 1.

[125] R. K. Schmidt and T. Leinmüller, "A spatio-temporal metric for the evaluation of cooperative awareness," in *18th World Congress on Intelligent Transport Systems*, 2011.

[126] R. Barr, Z. J. Haas, and R. van Renesse, *Scalable Wireless Ad hoc Network Simulation*. CRC Press, Aug. 2005, ch. 19, pp. 297–311. [Online]. Available: http://www.amazon.com/Handbook-Theoretical-Algorithmic-Wireless-Networks/dp/0849328322

[127] D. R. Choffnes and F. E. Bustamante, "An integrated mobility and traffic model for vehicular wireless networks," in *Proceedings of the 2nd ACM international workshop on Vehicular ad hoc networks*, ser. VANET '05. New York, NY, USA: ACM, 2005, pp. 69–78. [Online]. Available: http://doi.acm.org/10.1145/1080754.1080765

[128] E. Schoch, M. Feiri, F. Kargl, and M. Weber, "Simulation of ad hoc networks: ns-2 compared to jist/swans," in *First International Conference on Simulation Tools and Techniques for Communications, Networks and Systems (SimuTools 2008)*, Marseilles, France, Mar. 2008.

[129] SAE International, "DSRC Implementation Guide - A guide to users of SAE J2735 message sets over DSRC," Tech. Rep. v20, February 2010. [Online]. Available: http://www.sae.org/standardsdev/dsrc/DSRCImplementationGuide.pdf

[130] J. Freudiger, M. Raya, M. Félegyházi, P. Papadimitratos, and J.-P. Hubaux, "Mix-Zones for Location Privacy in Vehicular Networks," in *ACM Workshop on Wireless Networking for Intelligent Transportation Systems (WiN-ITS)*, Vancouver, 2007.

[131] R. K. Schmidt, A. Brakemeier, T. Leinmüller, F. Kargl, and G. Schäfer, "Advanced carrier sensing to resolve local channel congestion," in *Proceedings of the Eighth ACM international workshop on Vehicular inter-networking*. ACM, 2011, pp. 11–20.

[132] J. Y. Petit, M. P. Feiri, and F. Kargl, "Spoofed data detection in vanets using dynamic thresholds," in *Proceedings of the IEEE Vehicular Networking Conference (VNC 2011), Amsterdam, Netherlands*. USA: IEEE Communications Society, November 2011, pp. 25–32.