

UNIVERSITAT ROVIRA I VIRGILI

Universitat Rovira i Virgili

Department of Computer Engineering and Mathematics

Ph.D. Dissertation

Enhancing Drivers' Privacy in Urban Electronic Road Pricing Systems

Author:

Roger JARDÍ CEDÓ

Thesis Advisors:

Dr. Jordi CASTELLÀ-ROCA and Dr. Alexandre VIEJO

Dissertation submitted to the Department of Computer Engineering and Mathematics in partial fulfillment of the requirements of the degree of Doctor of Philosophy in Computer Science

Roger Jardí Cedó

Enhancing Drivers' Privacy in Urban Electronic Road Pricing Systems

Ph.D. Dissertation

Directed by

Dr. Jordi Castellà-Roca and Dr. Alexandre Viejo

Department of Computer Engineering and Mathematics

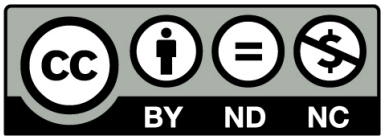


Universitat Rovira i Virgili

Tarragona

2015

© This work has been carried out by Roger Jardí Cedó, 2015, under the Creative Commons license of the type Attribution-NoDerivativeWorks-NonCommercial [1].



[1] To view a copy of this license, please visit:
http://creativecommons.org/licenses/by-nc-nd/3.0/

UNIVERSITAT ROVIRA I VIRGILI

DEPARTMENT OF COMPUTER ENGINEERING AND MATHEMATICS

I STATE that the present study, entitled "Enhancing Drivers' Privacy in Urban Electronic Road Pricing Systems", presented by Roger Jardí Cedó for the award of the degree of Doctor, has been carried out under my supervision at the Department of Computer Engineering and Mathematics of this university.

Tarragona, July 1, 2015

CPISR-1 C Jordi Castellà Roca

Firmado digitalmente por CPISR-1 C Jordi Castellà Roca
Nombre de reconocimiento (DN): c=ES, o=Universitat Rovira i Virgili, ou=Enginyeria Informàtica i Matemàtiques, ou=Serveis Públics de Certificació CPISR-1 amb càrrec, ou=Vegeu https://www.catcert.cat/verCPISR-1CarrecUR (c)03, sn=Castellà Roca, givenName=Jordi, title=PRF, serialNumber=43730310L, cn=CPISR-1 C Jordi Castellà Roca
Fecha: 2015.07.02 13:19:35 +02'00'

LUIS ALEXANDRE VIEJO GALICIA

Firmado digitalmente por LUIS ALEXANDRE VIEJO GALICIA
Nombre de reconocimiento (DN): c=ES, ou=Vegeu https://www.catcert.cat/veridCAT (c)03, ou=Serveis Publics de Certificacio CPIXSA-2, sn=VIEJO GALICIA, givenName=LUIS ALEXANDRE, serialNumber=39710359P, cn=LUIS ALEXANDRE VIEJO GALICIA
Fecha: 2015.07.02 13:40:30 +02'00'

Dr. Jordi Castellà-Roca
and
Dr. Alexandre Viejo,
Doctoral Thesis Supervisors

Approved by the University Committee on Graduate Studies:

Tot el treball dut a terme durant aquests últims anys culmina amb aquesta tesi. En aquest període he viscut molt bons moments, bonics, feliços, i sobretot de molt aprenentatge. Tot i això, també n'hi ha hagut de molt difícils, en els que la complexitat d'aquest treball, malauradament, es barrejava amb els dolorosos moments de la malaltia del meu pare. Per aquests bons moments viscuts i per haver fet més suportables aquells més difícils, m'agradaria donar les gràcies a totes les persones, que directa o indirectament, m'han ajudat i recolzat.

Jordi i Alex, us agraeixo tot el que m'heu ensenyat i ajudat durant aquest temps, i especialment, el suport que m'heu donat fora del treball.

Vull agrair a la Magdalena Payeras i al Macià Mut de la Universitat de les Illes Balears per la seva col·laboració, i als membres del CRISES i companys de laboratori, en particular, a la Tamar Molina pel suport lingüístic, a l'Aida Calviño pels seus consells, al Josep Ma. Mateo pel seu assessorament, al Guillem Rufián per la seva ajuda, i al Josep Domingo i Jesús A. Manjón pel seu suport.

També m'agradaria agrair a tota la meva família i amics per haver-me fet costat. Molt especialment a la meua mare i al meu germà per tot el que han fet durant aquest temps, tant per mi com per tota la família, perquè tot junts segur que ens en sortirem.

I a tu Sabina, que a part de ser partícip, has pogut viure-ho tot de ben a prop, ningú millor que tu sap tot l'esforç que ens ha costat.

Pare, tot i que no hi ets ara mateix, has pogut veure com hi arribava.
Et trobo molt a faltar.

Abstract

Over the last century, vehicles have become the means of transport *par excellence*. The widespread vehicle adoption by our societies has been a revolution in terms of employment patterns, social interactions, convenience and economy. They undoubtedly bring many benefits, but they also entail some drawbacks such as an increase of traffic congestion, accidents, air and noise pollution, or de-ruralization.

Recently, the aggravation of these problems in urban areas, the emergence of the Information and Communication Technologies, and a greater awareness in society of these problems have led to solutions such as the deployment of Electronic Road Pricing (ERP) systems. The main purpose of these systems is to restrict the access of vehicles to certain city areas, named Low Emission Zones (LEZ), for which a toll is assessed according to traffic conditions and vehicle emissions.

Since their adoption, these solutions have proven to be quite promising. However, current proposals are still far from being ideal. While reducing congestion to some extent, they exhibit several shortcomings, above all, related to payment fairness and privacy issues. More specifically, current schemes still introduce a significant error percentage in the detection of fraudulent drivers. Moreover, they usually require toll systems to be equipped with cameras that take pictures of all the vehicles that pass through the control points. This behavior may represent a serious privacy threat for drivers.

This thesis aims at providing security and privacy to new urban Road Pricing Systems. More precisely, the existing systems have been analyzed from a fraud and a privacy points of view. From its conclusions, two new Electronic Road Pricing Systems for Low Emission Zones have been proposed with the aim of detecting fraud while preserving drivers' privacy. In particular, they provide a deterministic fraud control and revocable anonymity for vehicles that misbehave. Both proposals allow to disperse traffic from areas with high traffic density with the aim of reducing the pollutant emissions by reducing traffic jams and time spent. However, the way to address the problem is different in each proposal. In the first proposal, drivers pay depending on the duration of the stay in the LEZ. In the sec-

ond proposal, drivers pay according to the path they have covered in the LEZ. They include a security and privacy study, which shows that privacy is preserved while fraud is detected, and a deployment feasibility study that shows that the provided proposals are realistic and may be deployed in practical scenarios.

Contents

1 Introduction **1**

1.1 Motivation 1

1.2 Contributions of this Thesis 4

1.3 Structure of this Document 6

2 State of the Art **7**

2.1 Related Work 7

2.1.1 VPriv [1] 8

2.1.2 A group signature based electronic toll pricing system [2] 9

2.1.3 Privacy-friendly electronic traffic pricing via commits [3] 10

2.1.4 Privacy enhanced fraud resistant road pricing [4] . . . 10

2.1.5 PrETP [5] 10

2.1.6 Milo [6] 12

2.1.7 SPEcTRe [7] 12

2.1.8 Cell-based privacy-friendly roadpricing [8, 9] 14

2.2 Analysis of System Features 14

2.3 Conclusions of the State of the Art 16

3 Fundamentals of the Proposals **19**

3.1 Dealing with Fraud and Privacy: Authentication with Revocable Anonymity 19

3.2 Privacy Authentication with Revocable Anonymity Scheme . 21

3.2.1 Certification Setup 23

3.2.2 Certificate Authority Installation 24

3.2.3 Certificate Generation 24

4 Time-based Electronic Road Pricing System for Low Emission Zones Preserving Drivers' Privacy **25**

4.1 Novelty of the Approach 26

4.2 System Model 27

4.2.1 Actors Involved 28

4.2.2 Requirements 29
4.2.3 Adversary Model 30
4.3 General Overview 31
4.4 Entrance/Departure LEZ Protocol 32
4.4.1 Setup 33
4.4.2 Certification 35
4.4.3 Certificate Generation 35
4.4.4 Check-in 36
4.4.5 Check-out 38
4.4.6 Extending the Protocol: Multiple Zones 40
4.4.7 Extending the Protocol: Addressing Residents 42
4.5 Payment Protocol 43
4.5.1 Price Generation 43
4.5.2 Payment 43
4.5.3 Payment Verification 44
4.5.4 Extending the Protocol: Multiple Zones 45
4.5.5 Extending the Protocol: Addressing Residents 46
4.6 Sanction Protocol 47
4.6.1 Sanction 48
4.6.2 Extending the Protocol: Multiple Zones 49
4.6.3 Extending the Protocol: Addressing Residents 50
4.7 Security and Privacy Analysis 51
4.8 Functional Requirements Analysis 55
4.8.1 Online-feasibility Study 55
4.8.2 Study of the Electronic Payment System Adaption . . 60

5 Privacy-Preserving Electronic Road Pricing System for Low Emission Zones with Dynamic Pricing 65
5.1 Novelty of the Approach 66
5.2 System Model 67
5.2.1 Architecture and Participants 67
5.2.2 Requirements 69
5.3 General Overview 71
5.4 Setup Protocol 75
5.5 Certification Protocol 76

5.5.1 Certificate Entity Installation 76
5.5.2 Certificate Generation 76
5.6 Payment Protocol 76
5.6.1 Price Generation 77
5.6.2 Ticket Acquisition 77
5.7 Entrance Protocol 80
5.8 Off-line Protocol 81
5.8.1 Doublespending Verification 81
5.8.2 Sanction Process 82
5.9 Security and Privacy Analysis 85
5.9.1 Adversary Model 85
5.9.2 Security Analysis 87
5.10 Functional Requirements Analysis 96
5.10.1 On-line Feasibility Study 97
5.10.2 Traffic Simulation 102

6 Conclusions **109**
6.1 Contributions 109
6.2 Publications 112
6.3 Future work 112

Chapter 1

Introduction

In this chapter, the scope and the main issues addressed in this dissertation are introduced in §1.1. The contributions to the field are then briefly described in §1.2. Finally, the structure and organization of this thesis are presented in §1.3.

Contents

1.1 Motivation **1**

1.2 Contributions of this Thesis **4**

1.3 Structure of this Document **6**

1.1 Motivation

In recent years, traffic congestion has become a significant problem for almost all major cities in the globe. The center of major metropolitan areas have a huge vehicle concentration in some specific locations, usually in the entrance and exit points to the city. The concentration of so many vehicles commonly causes traffic jams and cause circulation problems.

Traffic congestion has a clear negative impact on their citizens. Nowadays, drivers waste a lot of time in their vehicles, for example, when getting to work, and they consequently suffer stress in many occasions. But these are not the only problems that derive from traffic congestion. Many citizens, who cope with traffic congestions every day as bystanders, suffer from air and noise pollution produced by vehicles. Motor vehicles are a significant source of urban air pollution. However, traffic pollution is originated not only from the combustion particles, but also from road dust that originates from the wear of road surfaces, brakes, clutches and tires. Traffic emissions are the main source of intra-urban variation in the concentrations of air pollutants in many cities [10]. There are a number studies [11, 12, 13] that show higher levels of pollutants in proximity to roads, and others studies

[14, 15, 16] have shown an excess of health risks in proximity to roads. It is also corroborated in the Review of evidence on health aspects of air pollution, presented by OMS in 2013 [17]. They found evidence of increased health effects linked to traffic emissions.

As a consequence, several organizations and governments have focused their efforts on reducing air pollution. An example of this is the air quality guidelines presented by OMS in 2005 [18], which give guidance on "the way to reduce air pollution effects on health". Based on these recommendations, different European directives, such as 2008/50/CE, limit the level of certain environmental pollutants. The European Commission has recently adopted the Clean Air Policy Package, which includes a new strict directive to reduce pollution, and a Clean Air Programme for Europe with measures to ensure that the new quality objectives for the period up to 2030 are met.

In order to solve this situation, or at least, reduce air pollution harmful effects and fulfill this legislation, governmental and private institutions have attempted to find suitable solutions that are mainly based on discouraging people from using their own vehicles. Well-known measures resulting from these efforts are providing better public transportation services, by introducing new taxes to the owners of vehicles, High-Occupancy Vehicle (HOV) lanes [19], variable speed city entrances, or by implementing toll systems in centric urban areas, also known as *low emission zones*. In [20], authors assert that low emission zones can substantially reduce local levels of traffic-based air pollution (particulate matter from traffic dropped by 60%) after the development of a low emission zone in Munich.

Among these systems, the use of toll systems has received a lot of attention. **Low-Emission Zone (*LEZ*)** has been adopted in many cities such as London, Singapore, Tokyo or Beijing [21, 22, 23]. The main reason behind the success of this method is that it enables an authority to restrict the access of vehicles for which a fee is assessed according to certain conditions, such as weight or emissions. In this way, only those drivers that drive in a certain area of the city have to pay for this concept. This method is especially famous in Europe, where it is often used for preserving historical city centers (i.e., Athens or London).

More recently, the European Commission, apart from regulating the limit of pollutant emissions, also works towards this outcome through the CLARS

platform (Charging, Low Emission Zones, other Access Regulation Schemes) due to the novelty of low emission zones. The aim of this platform is to help and provide information and support, as well as a public information website[1], to public authorities to consider the development of a *LEZ* as a measure to improve environmental quality and lower traffic-related health risks in their cities. As this site shows, there are some different variations of *LEZ* depending on the number and the distribution of the zones. The city of Rome has a *multiple LEZ*. Specifically, it has four nested zones; the outer zone is composed of four adjacent subzones between them. What's more, there are cities with **Zero-Emission Zone**, where only zero-emission vehicles are allowed.

Since some decades ago, Electronic Toll Collection (ETC) has been used in highways, tunnels or bridges in order to expedite toll payments as well as to reduce traffic jams. E-ZPass in USA, AutoPass in Norway or Via Verde in Portugal are examples of these electronic systems, which try to reduce the delay and vehicle congestions in toll areas. Likewise, thanks to new technologies such as the GPS and wireless communication, vehicular location-based services (VLBS) have been developed with the purpose of providing information to drivers in relation to their geographic location, and improving transportation efficiency. These ETC systems, considered VLBS, are known as **Electronic Road Pricing (ERP)**. *ERP* systems work in a fully unsupervised way. In this way, Singapore can be considered the leading example as it has implemented an electronic road pricing system since 1998, with variable prices [21]. Similar approaches are being gradually adopted worldwide. For instance, they are already used in Washington, Georgia, Virginia and Toronto. In addition to reducing the traffic, *ERP* systems can use variable prices to manage the flow of vehicles and, hence, the traffic density. More specifically, the traffic authority can increase the toll taxes in congested roads and suggest drivers to take cheaper routes.

It has been acknowledged that these already deployed toll systems, while reducing congestion to some extent, exhibit several shortcomings related to payment fairness, privacy issues, congestion distribution to other routes or lack of proportionality. One of the most serious problems affects the fraud control effectiveness and the drivers' privacy. These problems arise from

[1] `http://urbanaccessregulations.eu/`

the system operations. On the one hand, the use of "spy cameras" like in the London LEZ case, could lead to an improper registration of fraudulent vehicles, although some of them would be authorized vehicles. On the other hand, the registration of the vehicle locations, and also the extensive use of cameras, could facilitate the monitoring of drivers, thus turning *ERP* systems to non-anonymous systems. As a consequence, the scientific community has recently focused on this topic in order to design new *electronic road pricing* systems that address these drawbacks. The most relevant *ERP* systems in the literature are [1, 2, 3, 4, 5, 6, 7, 9].

1.2 Contributions of this Thesis

A *LEZ* is an urban area that restricts the access of *V*s in exchange for the payment of a tax in order to reduce the pollutant emissions and improve traffic problems. In this environment, *Electronic Road Pricing* systems are of great help. This dissertation is focused on *Electronic Road Pricing* systems in *Low Emission Zones*. This work especially studies the privacy of drivers and fraud control mechanisms of *LEZ* systems.

One of the objectives of this document is to present a survey on the current ERP systems proposed in the literature. Despite of the fact that these systems have not particularly been designed for urban zones, they could easily be adopted in this environment. This state of the art includes a classification and analysis of advantages and disadvantages of the different kinds of approaches.

This work also contributes to *ERP* systems by offering two new proposals. Based on the study of the current literature, some points of improvement such as fraud control effectiveness, drivers' privacy and feasibility of the deployment of ERP systems in real environments, have been identified and have been taken into account in two new schemes in order to provide:

- A non-probabilistic fraud control based on the identification of fraudulent vehicles and their identification by means of two methods: (i) revocable anonymity of dishonest drivers that collaborate with the system, and (ii) registration of dishonest drivers that do not collaborate with the system with *Checkpoints*. These *Checkpoints*, which are deployed in the streets by the *Service Provider* in order to control the access to the

LEZ by vehicles, are equipped with cameras that only take pictures of the vehicles that misbehave.

- Privacy for honest drivers. Only dishonest drivers are identified through revocable anonymity or registration by *Checkpoints* and thus, they are affected by a loss of privacy. Otherwise, honest drivers keep their privacy because they are not identified. For this reason, drivers are expected to collaborate with the system to keep their privacy. That is, if a driver wants to keep her privacy, she should then behave correctly and cooperate. In this way, accurate routes of honest drivers are not obtained and their privacy is preserved. Moreover, *OBUs* do not register vehicles' geolocations.

- Efficient cryptographic primitives are used in these systems in order to effectively accomplish with the two previous improvements, fraud control and drivers' privacy.

Both proposals are designed with the aim to accomplish with the purpose of *LEZ*s, that is to decrease pollutant emissions. The way in which both solutions try to disperse traffic from areas with high traffic density is different. Addressing this problem differently has resulted in two systems with different features. As far as the first one is concerned, a greater user privacy is provided since this system is not aware of the movements of vehicles in the *LEZ*. Regarding the second one, it provides more precision in the pricing of the *LEZ*, which could facilitate the management of traffic more accurately as users are suggested to take less expensive itineraries. However, a increase in infrastructure investment is required, because stretches of streets are controlled by the *ERP* system. The choice between one or the other will depend on the requirements that the system provider wants to obtain.

The first proposal is based on a time approach. In this system, drivers pay depending on the duration of the stay in the *LEZ*. This proposal allows to suggest drivers to cover short itineraries in the *LEZ*, and even to dissuade the entrance to the *LEZ* by drivers. This results in a *LEZ* composed of a unique zone. However, this scheme can also consider a *LEZ* with multiple zones. In this case, a further discrimination of prices is obtained according to the fragmentation the *LEZ*. This system follows a post-payment approach

as, a priori, it is not possible to know how long a vehicle could circulate in the *LEZ*.

The second proposal is based on a path approach. In the system proposed, drivers pay according to the path they have covered in the *LEZ*. Unlike the first proposal, this one allows the distribution of the traffic in the *LEZ*. This is achieved by means of the dynamic assignment of prices for each stretch. As the traffic conditions are constantly changing, the prices are periodically adapted to such conditions. In such a way, prices are computed according to the traffic density of the stretch. Then, stretches with high occupancy are more expensive. Therefore, *D*s are suggested to re-plan their route in order to cover less expensive routes, which depend on the distance to be covered, the consumption of fuel and the prices of the stretches. In this proposal, the payment of the tolls follows a pre-payment approach, even though a post-payment approach could be used.

1.3 Structure of this Document

Chapter 2 surveys and classifies the different existing ERP systems. Each proposal is described and its main advantages and disadvantages are discussed. The basics of the two proposals are presented in 3. The contributions to the field are described in Chapter 4 and Chapter 5. In particular, Chapter 4 presents a new *ERP* system based on a time approach. Chapter 5 presents another new *ERP*, which is based on a path approach. Finally, Chapter 6 includes the conclusions of the work and the main lines of future research are described.

Chapter 2

State of the Art

In recent years, several electronic systems dealing with the management of traffic depending on toll taxes, have been proposed in the literature. The most important ERP *systems are described in §2.1. They are then analyzed and evaluated according to their fraud control mechanisms and the level of drivers' privacy in §2.2. Finally, the conclusions of the previous analysis are stated in §2.3.*

Contents

2.1 Related Work 7

2.1.1 VPriv [1] 8

2.1.2 A group signature based electronic toll pricing system [2] 9

2.1.3 Privacy-friendly electronic traffic pricing via commits [3] 10

2.1.4 Privacy enhanced fraud resistant road pricing [4] . 10

2.1.5 PrETP [5] 10

2.1.6 Milo [6] 12

2.1.7 SPEcTRe [7] 12

2.1.8 Cell-based privacy-friendly roadpricing [8, 9] . . . 14

2.2 Analysis of System Features 14

2.3 Conclusions of the State of the Art 16

2.1 Related Work

A description of the most important *ERP* systems that can be found in the literature are detailed below:

2.1.1 VPriv [1]

VPriv provides a practical protocol to compute path functions for several driving-related problems while maintaining a high level of privacy.

In the registration phase, drivers generate a set of commitments containing different random TAGs previously generated. Ciphertexts of these commitments are sent to the service provider and these are bound to the driver's identity. However, TAG values are not known by the service provider in this phase.

In the driving phase, the vehicle sends time-location tuples together with a different valid TAG every period of time to the service provider. Then, the service provider does not know which car uploaded a certain tuple. To ensure users uploading the correct data, the authority is required to randomly record some license plate, location, time tuples, and then challenges drivers with these records during the payment. It is important that the number of random observations or checkpoints are kept to a moderate amount to maintain drivers' privacy.

At the end of the billing period, the driver has to pay according to the path driven and has to prove that the tuples uploaded by her are consistent with these checkpoint evidences. The server then computes and returns all location fees. Each user adds up her location fees according to her tags and proves the correctness of the sum to the server by using zero-knowledge proof, without revealing the ownership of the tags. This process needs to run several rounds every time to avoid user behaviors deviating from the system. This protocol is possible thanks to the use of additive pricing functions which support the use of homomorphic commitments whereby the drivers commit to the prices for each segment of their path as well as the sum of the prices.

Therefore, the product of the commitments is a commitment of the sum of the prices. This eliminates the need for a protocol to verify that the sum of segment prices was computed correctly.

Even though this protocol uses a secure two-party computation to make sure drivers cannot cheat on the total tolling price, this computation and communication overhead increases linearly with the number of rounds executed and with the number of users. In addition, the communication es-

tablished between drivers and the service provider has to be anonymous in order to upload the road segments they drove. To avoid linking drivers to their IP address, they must use an anonymizing network such as Tor.

Additionally, this system tries to solve other issues like speeding detection and insurance pricing.

2.1.2 A group signature based electronic toll pricing system [2]

This system is based on VPriv's idea of computing fees in the server from the time-location data of vehicles. Unlike other systems, the vehicles are divided by a trusted authority into groups. For every fixed period, the OBU of each vehicle signs its location records using a group signature scheme before and sends them to the toll server. In each billing period the toll server sends each driver all the attached location data associated to her group, together with the fee of each record. According to the local records of her own location tuples, the client adds up the fees and sends the sum to the server. After adding all client payments, the server checks whether the sum equals the sum of all location fees. If this is the case, the server collects the correct amount of tolls and spots the protocol. Otherwise, clients could be making incorrect payments. The group manager then determines the misbehaviors by opening location signatures, and computes the real toll payment for each client.

The group signature scheme preserves client anonymity in groups. That is, other entities won't be able to learn who in the group signs the message. Only the group manager has the ability to reveal the signer's identity. The toll server computes the fees by collecting anonymous location records from clients on roads, and by returning each cost of location to clients during toll calculation.

However, this system does not have any other fraud control mechanism that considers other possible system malfunctions. They assume"that all data generated by OBUs are reliable and trustworthy" and "OBUs are issued by the authority and built according to modern tamper-resistant techniques". They also consider that "the only way to generate false location tuples is signal spoofing which is not practical as signal simulators are available at a cost that significantly outweighs the saving of toll payment". Another assumption made by authors is that "location tuples are transmitted

to the toll servers anonymously". A part from that, this systems do not use either fraud control mechanisms such as random observations while driving, or audit processes for testing the time-location data sent by vehicles according to the checkpoint photographs at the billing period.

2.1.3 Privacy-friendly electronic traffic pricing via commits [3]

In this system, similarly to VPriv, the server collects location data. In this case, hash values of the trip records are used as location data. Drivers commit to the path they drove without revealing the individual road segments. The system uses hash functions for commitments, thus making it very efficient. Only additive road pricing functions are allowed (i.e., functions for which the cost of driving along a path is the sum of the cost of driving along each segment of the path); this allows a protocol to verify that the total fee has correctly computed as the sum of each road segment price by revealing a path from the root to a single leaf in a Merkle hash tree. In addition, this system uses checkpoints to verify that the driver faithfully reported each road segment on which she drove.

2.1.4 Privacy enhanced fraud resistant road pricing [4]

This article notes that in systems such as VPriv or PrETP, the audit protocol allows the government to query cars about locations where there is no camera, a capability that could be misused, for example, to identify whistleblowers. They propose a privacy-preserving tolling system in which vehicles can be spot-checked only in places where their presence is actually recorded, and in which overall driver privacy is guaranteed as long as the pricing provider and aggregation provider do not collaborate. Like VPriv, Hoepman and Huitem's system requires road segments to be transmitted from the car to the authority over an anonymizing network.

2.1.5 PrETP [5]

PrETP proposes the use of a trusted tamper-resistant hardware in each car to locally calculate the fees according to the vehicle path. While driving, the on-board unit of the vehicle collects location data and slices them in

segments (loc,time). For each collected segment, the OBU commits to each one.

This fact requires the use of a cryptographic protocol, called Optimistic Payment, in order to prove to the service provider that these calculations are carried out correctly. The Optimistic Payment, which is based on homomorphic commitments and signature schemes that allow zero-knowledge proofs of signature possession, allows on-board units to send the service provider the final fee commitments to the locations and prices used in the fee computation. These commitments do not reveal information on the locations or prices to the service provider. Moreover, they ensure that drivers cannot claim that they are located in any other place or that have used different prices to those used to create the commitments.

At the end of the billing period, the OBU generates, signs and sends the service provider a payment message containing the total fee computation and the commitments of the segments. The service provider then verifies the signature and checks that the aggregation of all sub-fees is well done. In addition, in order to check the veracity of the committed values, the service provider has access to a proof such as a photograph placing a vehicle in a specific point at a particular time. The service provider challenges the corresponding on-board unit by requesting a proof, thus proving that the location point where it is spotted is correct and that it is used in the calculation of the final fee. The on-board unit must open the commitment containing these locations, thus revealing only the location data and the price at the moment specified in the proof. For each client, only the total payment amount and location-time tuples in the physical vicinity of random checkpoints are revealed.

PrETP addresses some of the shortcomings in VPriv. In PrETP, drivers do not reveal the road segments they drive in the clear, so they do not need an anonymizing network. Instead, they commit to the locations and to the prices used in the fee computation; in the audit protocol, they open the commitments corresponding to the road segments where checkpoint cameras observed them. However, the lack of location data collection may deprive the system from the chance to assist the government with policy making for public interest, or to solve legal issues when needed.

Based on the same idea of PrETP, the same authors present a pay-as-you-

drive insurance system called PriPAYD.

2.1.6 Milo [6]

Milo is a modification of the PrETP system that has the aim to improve effectiveness of fraud detection, which depends on the ignorance of the location and the number of checkpoints by drivers. This drawback is not a specific problem of the PrETP system, but it is shared by others, such as VPriv, based on the idea of controlling fraud detection with the use of random checkpoints. In particular, this system fights against the possibility of drivers colluding to learn the spot-checking locations. This solution uses blind identity-based encryption to avoid revealing these locations to drivers. However, this article concludes that in practice, there is a gap between the assumption that checkpoints are unpredictable and the use of real-world cameras used to implement them. In addition, an increase in the amount of checkpoints affects the drivers' privacy: "If drivers are able to avoid some cameras, more of them will be required; if too many spot-check cameras are deployed, the records they generate will themselves degrade driver privacy. We believe that it is important for working on privacy-preserving tolling to address this limitation by carefully considering how the spot checks it relies on will be implemented".

2.1.7 SPEcTRe [7]

SPEcTRe improves the PrETP implementation by reducing the number of tolling points while maintaining driver honesty and privacy. It is the first system to use e-cash (by Chaum or others) to pay for driving. In VPriv, PrETP and Milo, spot-checks are necessary to uncover dishonest drivers. SPEcTRe maintains the same level of privacy as these other schemes for drivers, in the sense that in all these schemes, location-time information of a vehicle is only revealed at checkpoints. Their authors present two variants of the system: the spot-record scheme and the no-record scheme.

In the first scheme, the driver has a set of pre-generated tokens which are signed by the service provider and which include a correspondence with the license plate. During the driving, the driver broadcasts tokens in a certain period. A verifier secretly monitors drivers at random locations and times.

For each driver, the verifier takes a picture of the driver's license plate and records all tokens broadcasted. Each photograph has a set of tokens associated. The service provider then verifies that the signatures are valid for each picture, and that there are no repeated signatures. At the end of the billing period, the driver interacts with the payment server to redeem unused tokens, and she pays for the tokens used. The service provider verifies that tokens have not been collected by checkpoints. The possibility to pay for the tokens at the generation phase is proposed as an alternative to the post-payment scheme. The effectiveness of the system is based on the ignorance of the checkpoint time-location by drivers. If it is discovered, the system fails. In the same way as other systems, such as PrETP or Vpriv, the increase in the number of checkpoints affects the privacy of drivers who always take the same routes at the same time.

In the second scheme, the use of a wireless communication technology, which is able to determine which car is sending which token (directional antennas or triangulation), is assumed. The driver obtains a set of eCash tokens from the system provider at the beginning. While driving, the driver broadcasts tokens at a predetermined rate. The verifier checks the tokens. If the signature of the token is not valid, or the driver is not broadcasting tokens all the time, the verifier takes a photograph. In this point, the system is based on the good will of the verifier. At the end of the billing period, the tokens are invalidated, and the driver then interacts with the payment server to redeem unused tokens, and she pays for the tokens used. The invalidation can be done by using different public keys for each period of time. Double spending is an important issue addressed by the system in order to maintain a sorted and unified (centralized) list of detected tokens. If the checkpoint location is discovered, the effectiveness of the system also fails, and tokens from honest drivers can indeed be stolen. However, the privacy of the driver is guaranteed if the token generation is made through an anonymous channel.

In both variants, as in the rest of the systems of the literature, even assuming that drivers cannot detect the checkpoints, the detection of fraudulent drivers is not completely guaranteed. It depends on the number of checkpoints. In this case, it also depends on the fraction of spent tokens which are detected by verifiers. In addition, they have neither intensive nor

interactive computation on OBU.

2.1.8 Cell-based privacy-friendly roadpricing [8, 9]

In this system, a road pricing area is fixed by a square of cells with a relatively small size. When a vehicle enters a cell, it has to pay a certain amount of money. In this case, the fee computation depends on the covered cells. A certain subset of cells are marked as checkpoints. They contain a hidden camera, which is placed randomly for a certain period of time. The secure element (SE) of the OBU knows the location of the camera checkpoints. The OBU then notices the SE when the vehicle enters a cell. The SE generates a ticket whose content depends on whether the cell has a checkpoint or not. This information will be later useful for fraud detection. The SE is trusted so it can simply accumulate the fees per cell, store the result in this non-volatile memory, and report the cumulative fee to the service provider at the end of the reporting period. The service provider trusts the accumulation of fees reported by the SE, although it does not believe that the SE has been notified appropriately by the OBU at every cell transition. The service provider then checks the tickets provided by the OBU in the reporting period. The service provider can open the tickets for the checkpoint cells where the vehicle has been spotted, and it can check whether the content is appropriate. This system also provides a parking and insurance solution by following the main idea.

2.2 Analysis of System Features

In this section, The most important features of these systems are analyzed, and they are then classified according to these features. This classification is outlined in table 2.1.

1. **Segment collection based on OBUs**: All these systems require that an on-board unit (OBU) is equipped in every vehicle in order to collect and send information related to the geographical position of the vehicle and the fees related to that specific path. Depending on the importance of the role played by the OBU in the system, the systems can be classified according to the use of:

- Thin OBU: [1, 3, 4] and [2] systems in which the OBU (or SE) of the vehicle commits itself to each step of the fee calculation and stores it in a database of users' travel history at server. After receiving the fee report, the service provider may ask the OBU to open certain commitments, corresponding to roadside camera locations, in order to check details of the cumulative report.
- Fat OBU: In [5, 6, 8, 9] and [7] systems, locations and tolls are managed by user devices while servers are allowed to process only aggregated data.

Both categories have advantages and disadvantages. Hiding locations from servers reduces concerns about location privacy drastically. However, the load for user devices is considerable. Typically, devices have to manage the storage of locations and try to convince servers to not have cheated, e.g., using of zero-knowledge proofs. On the other side, the availability of location databases collected by servers like in VPriv can help improve applications such as traffic monitoring and control although the integration of multiple systems should be carried out carefully, whereas solutions to preserve location privacy become a mandatory requirement.

2. **Broadband communication based**: All the systems described require the use of a broadband connection in the vehicle in order to intercommunicate information between vehicle's On-Board Units and SP server periodically. Commonly, in the systems based on thin OBU, registered vehicle paths are transmitted periodically. In the systems based on fat OBU, feed data are sent just at the end of each billing period control.

3. **Location based on GPS**. In all these systems, the OBU needs to obtain vehicle locations by using a GPS receiver in order to generate time-location tuples periodically.

4. **Post payment time**: In all the proposed systems in the literature, [1, 3, 4, 2, 5, 6, 7, 8, 9], fees are computed from the path covered by the vehicle at the end of the billing period. The payment is performed afterwards according to such fees, i.e. monthly. This mode requires

interaction or reconciliation between the driver and the system after service, at the billing period.

5. **Fraud Control based on random camera checkpoints**: In all these systems, the fraud is controlled by recording vehicle plates at various checkpoints placed randomly along the roads. These checkpoints (Chps), supplied with cameras, shoot every vehicle passing through in order to place them on record. After each billing period, the *ERP* system, with all the evidence taken by the checkpoints, challenges drivers to prove that their path contains the location of each checkpoint. The effectiveness of this mechanism depends strictly on the drivers' ignorance of the location of checkpoints.

	OBU Type		Broadband	GPS	Payment		Fraud Control
Systems	Thin	Fat	Comm.	Location	Pre	Post	based on Chp
[1]	✓		✓	✓		✓	✓
[2]	✓		✓	✓		✓	✓
[3]	✓		✓	✓		✓	✓
[4]	✓		✓	✓		✓	✓
[5]		✓	✓	✓		✓	✓
[6]		✓	✓	✓		✓	✓
[7]		✓	✓	✓		✓	✓
[9]		✓	✓	✓		✓	✓

Table 2.1: System Features

2.3 Conclusions of the State of the Art

In general, all *ERP* systems that can be found in the literature ([1, 2, 3, 4, 5, 6, 7, 9]) calculate road usage pricing by considering the vehicle itinerary, and they use on-board units (OBU), equipped in each vehicle, to record its path. OBUs are enabled with GPS and wireless communication capabilities; they periodically collect their geographical position; and they send it to a service provider (or similar) together with other relevant data.

Even though *ERP*s share common features, they mainly differ in the way the road usage fees are computed. In [1, 3, 4, 2], *OBU*s send information

of the covered path to an external server, property of a Service Provider (*SP*), which prices the itinerary according to its path in every billing period. Therefore, *SP* is in charge of calculating the fees in each billing period, according to the vehicle path. By contrast, in [5, 6, 9, 7], fees are calculated locally in each *OBU*, and they are then sent, as a unique sum, to the *SP* server in every pricing period. In this case, the information disclosure related to the vehicle location is minimal. In order to achieve it, cryptographic proofs are used to demonstrate that *OBU* has been honest in the fee calculus and aggregation.

Fraud control is an important objective to be accomplished by *ERP* systems. All these systems are designed to prevent any possible misbehavior by the drivers that would enable them to save money illegally. Disconnecting the *OBU* or modifying the flow of data generated by this device in any way are possible examples of misconduct. In order to avoid these situations, these proposals adopt control mechanisms based on the use of checkpoints (*Chp*s), with the aim to test their honesty. *Chp* are equipped with cameras and they are randomly located in the restricted areas. *Chp*s take pictures of all the vehicles that pass through them. In this way, their plate number is stored together with the different geo-positions and the exact time of each one. These three items allow the system to build a partial path of all the vehicles moving around the restricted area, and to verify that a certain driver has not altered the set of positions recorded by her car's OBU and provided to the *SP* during the billing period. It is worth mentioning that this approach for fraud detection has a certain failure probability that directly depends on the number of *Chp*s deployed in the restricted area, and it also depends on preventing drivers from ascertaining their exact position in advance.

In the evaluated systems, the achievement of a high level of drivers' privacy and the detection of fraud at the same time is a trade-off. When a high detection level is requested, privacy is affected and vice versa. Increasing the number of checkpoints that are deployed in a restricted area directly affects drivers' privacy due to the fact that, the more checkpoints there are, the bigger the set of registered real drivers' locations will be; and, therefore, the more accurate the drivers' paths will be. Note that drivers' privacy directly depends on the accuracy of the paths that the system can build by using drivers' recorded locations. Besides, if *Chp*s are randomly moved

every so often, and vehicle paths follow a routine (i.e. going to work), precision could even improve though privacy would be affected. Therefore, if the system knows the whole path of a certain driver with high accuracy, her whereabouts are no longer private. In this way, in order to preserve the drivers' privacy, the way checkpoints behave should be revised.

Furthermore, all these proposals rely on the wrong assumption that checkpoint locations are unpredictable by drivers, and this issue makes those schemes unable to effectively control drivers potential misbehavior (or fraud), as it is stated in the work presented in [6]. This fact, in turn, represents a relevant problem that clearly limits the deployability of those schemes in real environments.

Chapter 3

Fundamentals of the Proposals

In this chapter, the basics, which are based on the two proposals of this dissertation, are established. In §3.1, the need for an authentication of users while preserving privacy is stated. Considering this requirement, an analysis of several authentication methods is then performed. Finally, an authentication scheme accomplishing with the fixed requirements, which will be used in the thesis' proposals, is presented in §3.2.

Contents

3.1 Dealing with Fraud and Privacy: Authentication with Revocable Anonymity 19

3.2 Privacy Authentication with Revocable Anonymity Scheme 21

3.2.1 Certification Setup 23

3.2.2 Certificate Authority Installation 24

3.2.3 Certificate Generation 24

3.1 Dealing with Fraud and Privacy: Authentication with Revocable Anonymity

In order to accomplish with *LEZ*s' goals, fraud has to be effectively controlled. For this reason, the access to *LEZ*s has to be controlled by the system, in which each user is verified to have fairly paid. This control could be performed using toll barriers. However, they usually slow down the traffic and can even cause traffic jams. A system without barriers is then preferred in order not to affect traffic at tolls. In this way, the system should facilitate the entry of all users into the *LEZ* by identifying the users who have not paid correctly. The offenders could then be sanctioned.

Conversely, honest users should not be identified, and their privacy should be preserved as long as they follow the rules. Contrariwise, the *ERP* system could learn other sensitive information apart from users' identity, such as users' location, itineraries, or even their workplace.

As a result, users are therefore encouraged to prove that they are legitimate users without disclosing their identity. Their identity would only be revealed in case of fraud.

Signatures and Pseudonyms are cryptographic techniques that could address these requirements, which are widely used in similar applications such as vehicular networks (VANETs) ([24, 25, 26, 27, 28, 29]).

Signatures offer authenticity, non-repudiation and integrity. However, signatures do not preserve signers' privacy on their own since their issuers are identified. Instead, group signature schemes ([24, 25, 26, 30, 31]) provide a member of a group to anonymously sign on behalf of the group, and can therefore be anonymously authenticated. In addition, these schemes offer revocable anonymity of the signers.

Group signatures are also widely used in vehicular systems with such purpose ([24, 25, 26, 27]). However, they cannot be accommodated in this work for several reasons: (i) considering that one of the aims of *ERP* systems is to reduce traffic jams, they require efficient operations. In this way, vehicles have to authenticate nonstop with the system. The computational cost of a signature and of the verification could be too high to achieve it ([30, 31, 32]), especially if they are performed in an On Board Unit; and (ii) secret keys, which are used in the signature, have to be protected from thefts or relinquish other users in order to authenticate correctly, since this could lead users to commit fraud. For this reason, it is convenient to use a tamper-proof device in order to securely store the keys and to carry out group signatures. The authors ignore the existence of tamper-proof devices supporting group signatures with affordable prices in the current market. While it is true that a custom-made Hardware Security Module (HSM) could accomplish with these requirements, it has a high economic cost. As a consequence, the use of this kind of hardware is not considered in this work due to its economical unfeasibility, bearing in mind that each vehicle must have one of these devices installed.

The use of pseudonyms is another technique widely used in order to

prevent users' anonymity ([33]). However, the privacy they offer is weak because they suffer from linkability, that is, several authentications from the same user can be connected. In *ERP* systems, it could be a serious privacy problem since different entrances of the same user in the *LEZ* could be linked, and sensitive information could then be inferred. In the literature, some proposals ([28, 34, 35, 29]) solve this issue by using a set of pseudonyms for each user. One different pseudonym can then be used each time. Even so, this solution presents a usability problem since the set of pseudonyms is finite. As the existence of methods solving this issue remains unknown to the authors, pseudonyms are neither considered in this scenario.

3.2 Privacy Authentication with Revocable Anonymity Scheme

As it has been stated above, non of the authentication methods mentioned above fulfill the requirements of the authentication that an *ERP* system needs. For this reason, an *authentication scheme* based on the RSA signature is proposed.

In this scheme, a set of groups of users is fixed. Each user is then registered in one of these groups, in which users are distributed homogeneously. For each group of users, a *Certification Authority CA* is generated. This *CA* is then transferred to each user of the group. In this way, users belonging to the same group share the same *CA*.

Each member of the group has access to her *CA*, which allows her to generate public key pairs and their public key certificates. In particular, each public key certificate:

1. Does not contain information of the user and any other identifying data. In this case, public key is then used as a pseudonym and users are able to generate new credentials whenever they need it. In this way, they can avoid linkability in each authentication.

2. Has an extension containing a user's identifier probabilistically encrypted with the public key of a trusted party. This party, which is

constituted by a set of m members, securely generates the asymmetric key pair with a threshold scheme ([36]). Therefore, the secret key is divided into m prices or shares. Hence, anonymity can only be revoked by this entity if at least t from m members collaborate with the decryption of the extension.

3. Does not contain any value that can be linked with previous certificates that belong to the same user. Neither can the issuer (CA) be used to link certificates belonging to the same user since there is a set of users sharing the same CA.

At this point, users are able to anonymously sign on behalf of the group, and so be anonymously authenticated. In case of fraud, at least t from m members could revoke users' anonymity.

In spite of this, users would be able to modify the ciphered extension in order to commit fraud without being identified. Consequently, they may authenticate with the system, but they may never be de-anonymized in case of fraud. For this reason, a *Secure Element SE*, which is a low-cost tamper-proof device that offers efficient RSA functionalities, is used in order to generate and keep users' credentials securely. In particular, each *SE* is provided by a trusted entity, such as a competent traffic authority, which registers users in a group and securely installs the corresponding *CA* in the *SE*. Up to this point, *SE*s are then able to securely and dynamically generate public key pairs and their public key certificates, including the extension with the ciphered identifier. They are also able to sign messages using these credentials. All these sensitive operations are performed in the *SE* and thus, users are not able to modify the certificate without being detected.

Hence, the authentication scheme (i) preserves the user's privacy whereas she collaborates with the system, since the used credentials do not contain identifying information of the user and they cannot be linked with other credentials belonging to the same user; (ii) offers revocable anonymity thanks to the use of an extension containing a ciphered user identifier, which can only be opened by a trusted entity; (iii) uses efficient cryptographic primitives (RSA), which provide non-repudiation, integrity and authenticity with the signature; (iv) protects the generation of the credentials, their storage and the execution of the mentioned cryptographic primitives from

attackers by using a *SE*.

Note that the two *ERP* systems proposed in this work use this authentication scheme and their design follows the same idea of underlying this authentication method in order to preserve users' privacy while they collaborate.

The generation of the certificates used in this authentication scheme is composed of three phases: *Certification Setup*, *Certificate Authority Installation* and *Certificate Generation*. In the first phase, *CA*s are generated for each group of users. In the second phase, a user is registered in one of these groups of users and a *CA* is installed in the *SE*. In the third phase, the *SE*, which has been previously initialized, generates new credentials.

3.2.1 Certification Setup

In this phase, a traffic-competent authority, such as *Vehicle Certification Authority VCA*, defines a set of groups of users and generates a different *CA* for each group. *VCA* then needs to:

i. Securely generate an asymmetric key pair (Pk_{VCA}, Sk_{VCA})

ii. Securely obtain a CA certificate $cert_{VCA}$, which has a chain length of 1, and a certificate repository of the authorities from competent authorities (i.e. transit authority)

iii. Define a set of vehicles $V = \{v_1, v_2, ..., v_{num_V}\}$, where $num_V = |V|$ is the number of vehicles

iv. Define a collection of sets $K = \{C_1, C_2, ..., C_{num_K}\}$ partition of V, where $num_K = |K|$, with $|C_i| = num_C$, $\forall i \in [0, num_K)$

v. Generate and associate a certification authority VCA_{C_i} to each element of the subset K ($C_1, ..., C_{num_K}$):

   v.i. An asymmetric key pair ($Pk_{VCA_{C_i}}$, $Sk_{VCA_{C_i}}$), $\forall i \in \{1, ..., num_K\}$

   v.ii. A CA certificate $cert_{VCA_{C_i}}$, $\forall i \in \{1, ..., num_K\}$, which has a certificate chain length of 0

3.2.2 Certificate Authority Installation

In this phase, a traffic competent authority, such as *Vehicle Certification Authority VCA*, registers a user in a group of users and installs the corresponding Certification Authority in a *SE*. The process is carried out from time to time according to the revocation and renewal certificate policies adopted. *VCA* then needs to:

1. Register V in an element of the subset K (in a C_i)
2. Download the following data in the *SE* by means of a secure channel:
    - A certificate repository of certification authorities
    - Identifying information of the vehicle V_{id} (i.e. its license plate) and its technical specifications (i.e. owner, power or emissions of pollutant gases).
    - The certification authority VCA_{C_i} associated to C_i consisting of $Pk_{VCA_{C_i}}$, $Sk_{VCA_{C_i}}$ and $cert_{VCA_{C_i}}$

3.2.3 Certificate Generation

In this phase, the *Secure Element* generates new credentials. This phase is performed whenever a user wants before the authentication. The *SE*, which has a certification entity VCA_{C_i} installed, then needs to:

1. Compute an asymmetric key pair (Pk_{V_q}, Sk_{V_q})
2. Generate a public key certificate $cert_{V_q}$ with the following features:
    - An extension $cert_{V_q}.idS$ containing the probabilistic encryption (i.e. by using OAEP padding [37], standardized later in PKCS #1v2 and RFC 2437) of the vehicle identifier V_{id} with the public key of *PA*: $cert_{V_q}.idS = Enc_{Pk_{PA}}(V_{id})$
    - An extension $cert_{V_q}.em$ containing the pollutant emission category (i.e. European Emission Standards) of the V

Chapter 4

Time-based Electronic Road Pricing System for Low Emission Zones Preserving Drivers' Privacy

In this chapter, an Electronic Road Pricing (ERP) system, which is designed specifically for cities with Low-Emission Zones, is proposed. The aim of this system is to detect fraud and to preserve drivers' privacy. This system is based on a time approach, in which drivers pay depending on the duration of the stay in the LEZ. The novelty of the system is introduced in §4.1. The system is modeled in §4.2. In §4.3, the scheme is overviewed. Afterwards, the scheme is divided in three protocols, Entrance/Departure Protocol (§4.4), Payment Protocol (§4.5) and Sanction Protocol (§4.6). Finally, security and privacy are evaluated in §4.7, and performance and feasibility are presented in §4.8.

Contents

4.1 Novelty of the Approach 26

4.2 System Model 27

4.2.1 Actors Involved 28

4.2.2 Requirements 29

4.2.3 Adversary Model 30

4.3 General Overview 31

4.4 Entrance/Departure LEZ Protocol 32

4.4.1 Setup 33

4.4.2 Certification 35

4.4.3 Certificate Generation 35

4.4.4 Check-in 36
4.4.5 Check-out 38
4.4.6 Extending the Protocol: Multiple Zones 40
4.4.7 Extending the Protocol: Addressing Residents . . . 42
4.5 Payment Protocol 43
4.5.1 Price Generation 43
4.5.2 Payment 43
4.5.3 Payment Verification 44
4.5.4 Extending the Protocol: Multiple Zones 45
4.5.5 Extending the Protocol: Addressing Residents . . . 46
4.6 Sanction Protocol 47
4.6.1 Sanction 48
4.6.2 Extending the Protocol: Multiple Zones 49
4.6.3 Extending the Protocol: Addressing Residents . . . 50
4.7 Security and Privacy Analysis 51
4.8 Functional Requirements Analysis 55
4.8.1 Online-feasibility Study 55
4.8.2 Study of the Electronic Payment System Adaption 60

4.1 Novelty of the Approach

In this chapter, a new *ERP* system for *LEZ*s is proposed. It is based on a time approach with the aim of providing fraud control and honest drivers' privacy through revocable anonymity. In this way, tolls are fixed according to the traffic density of the *LEZ*, and drivers then pay according to the time spent in it. As a result, drivers are suggested to cover short itineraries, and even to dissuade drivers from entering the *LEZ*. This system can also consider a LEZ with multiple zones, which provides a greater management of the traffic. This system follows a post-payment approach as, a priori, it is not possible to know how long a vehicle could circulate in the LEZ.

All the systems described in Section 2 ([1, 3, 4, 2, 5, 6, 7, 8, 9]) penalize the privacy of honest or dishonest vehicles by taking a photo of them in different places. The price they pay for circulating in this kind of systems is a loss of privacy. Otherwise, in the system proposed here, only dishonest drivers are

photographed and they are thus affected by a loss of privacy. In this case, the *Chp*s, who are also equipped with cameras, only register fraudulent vehicles, thus keeping honest drivers' privacy. For this reason, drivers are expected to collaborate with the system to keep their privacy, that is, if a driver wants to keep her privacy, she should then behave correctly and cooperate. She will otherwise loses it. Moreover, in the system proposed, the *OBU* of the vehicle does not register its location and fraud control is non-probabilistic.

The resulting scheme has been tested in order to assess the feasibility deployment of the parts of the system with high temporary restrictions, and the results show that that the provided proposal is realistic and may be deployed in practical scenarios.

The new *ERP* system, which is presented in this chapter, is supported by:

- Roger Jardí-Cedó, Macià Mut Puigserver, Magdalena Payeras-Capellà, Jordi Castellà-Roca, Alexandre Viejo "Electronic Road Pricing System for Low Emission Zones to Preserve Driver Privacy" In *11th International Conference Modeling Decisions for Artificial Intelligence (MDAI 2014)* pp. 1–13. 2014.

- Roger Jardí-Cedó, Macià Mut Puigserver, Magdalena Payeras-Capellà, Jordi Castellà-Roca, Alexandre Viejo "Time-based Electronic Road Pricing System for Low Emission Zones Preserving Drivers' Privacy" In *International Journal of Computer Communications* 2015. Under review.

- Roger Jardí-Cedó, Macià Mut Puigserver, Magdalena Payeras-Capellà, Jordi Castellà-Roca, Alexandre Viejo "Sistema de Telepeaje en Zonas Urbanas" In *XIII Reunión Espanola de Criptografía y Seguridad de la Información (RECSI'14). (XIII Spanish Meeting on Cryptography and Information Security)*. pp. 93–99. 2014.

4.2 System Model

This section presents a system that is modeled by describing the participating actors, the requirements to be met and the several and possibly opposite interests of the actors involved.

Key	Name	Description
D	Driver	Person who drive a V
V	Vehicle	Means of transport, which is driven by a D
OBU	On Board Unit	Device installed in each V, which executes the protocol in a V
SE	Secure Element	Tamper-proof device that executes sensitive operations in a V
SP	Service Provider	Manager of a LEZ
Chp	Checkpoint	Points of control deployed in the entrance/exit of a LEZ
VCA	Vehicle Certification Authority	The authority who certifies Vs
PA	Punishment Authority	The authority who evaluates incidences and punishes fraudulent Vs

Table 4.1: Entity Table

4.2.1 Actors Involved

The proposal considers different actors. *Driver D* is the person who drives a vehicle in a *LEZ*. *Vehicle V* is the means of transport registered by a unique *D*, who is the owner of the vehicle, yet the vehicle may be driven by several*D*s. *V* has an identifier (the vehicle plate) that connects it to the owner. *Secure element SE* is a tamper-proof security module installed by the competent traffic authority in each *V*. It performs all the sensitive operations needed to meet the security requirements. *On-board unit OBU* is installed in each *V*. It has more computational power and storage capacity than *SE*. The device connects *SE* with the user and performs the less sensitive protocol operations. It has a location capability (GPS). *Service Provider SP* offers an ERP service for urban areas thanks to a concession contract with the local public administration (i.e. City Council). This entity has the right to offer this service and is responsible for managing the system. *SP* installs the checkpoints in the restricted zone. *Checkpoint Chp* aims to control the access of vehicles that enter (Chps of entrance) or leave (Chps of exit) the zone. *Chp*s are considered trustworthy and will never take photographs indiscriminately as long as their behavior is optimistic. *Vehicle certification authority VCA* provides keys and certificates to *V*s. *Punishment Authority PA* is a trusted entity composed by different sub-entities or authorities. The collaboration of a minimum set of sub-entities allows them to know and reveal the identity of the owner of the *V* in case of fraud.

4.2.2 Requirements

The system requirements related with fraud, privacy, authenticity and technology, are described below in order to establish the foundations of the system.

Anti-fraud Requirements

When a *V* enters or exits a *LEZ* through a *Chp*, they obtain **proof-of-entrance** γ_i or **proof-of-departure** γ_o. This γ_i contains information to prove that a specific *V* enters the *LEZ* through a specific *Chp* at a specific hour. This *proof* is considered **valid** when it cannot be modified once generated without detection (*integrity*), when its issuers can prove that it is their generation (proofs' *authenticity*), and also when they cannot deny its authorship (*non-repudiation*). Each proof is **linked** to a *V* and a *Chp*. The link between a proof and a *V* guarantees that the token cannot be used by another *V'*, neither in a voluntary nor in an involuntary way. This avoids the *duplicity* of a proof when a fraudulent *V'* tries to use the same proof at the same time. *SP* assures that all the *D*s pay correctly. If this is not the case, *SP* identifies the offending *D*s and generates evidences to prove it. The **fraud** is committed by a *D*, and detected by the system, when *D* drives in a *LEZ without a* γ_i, *with an invalid* γ_i, *with a valid* γ_i *but associated with another V*, or if *she doesn't pay correctly* in the exit. A *SP* cannot **falsely accuse** an honest *D* of fraud (an honest *D* should not be defenseless). A false accusation takes place when a *SP* unjustly claims that a *V does not have a* γ_i, *has an invalid proof*, *has a valid proof belonging to another V*, or if *she doesn't pay correctly* in the exit.

However, both proofs, γ_i γ_o, can be used as evidences by the users to refute a false accusation.

Authenticity Requirements

At the entrance and at the exit of a *LEZ*,*V*s and *Chp*s exchange information. When the communication is established, both parts, *V* and *Chp* must prove its identity to the other part. This way, each one can be sure that the protocol is executed with the right entity. If this is not the case, this action must be reported.

Privacy Requirements

The fraud control executed by *SP* can endanger the privacy of the *D*s. In this case, the curiosity of *SP* could cause an excessive monitoring of the system or even could trace the itinerary of a specific *V*. With the aim of avoiding this excessive control over the *V*s by *SP*, the system must (i) assure the **anonymity** (the identity of *D* or *V* cannot be linked to any itinerary); (ii) avoid the **traceability** (*SP* mustn't know the itinerary of an unidentified *V*); and (iii) provide **revocable anonymity** to *D* (if a *D* commits fraud, *SP* needs her identity in order to punish her and for this reason, the identity is revealed).

Functional Requirements

The **communication** between *V*s and *Chp*s needs to (i) be *started* when a *V* is close to a *Chp*, (ii) let *Chp* communicate with the *nearest V*, and (iii) be carried out when a *V* is *moving*. The communication has to start when a *V* is detected near from a *Chp*, for example by means of the broadcast of the information required to establish the communication. This broadcast can be carried out by the use of Bluetooth Low Energy technology. Moreover, the communication in movement can be possible by combining low and medium distance communication technologies, such as Wimax, ZigBee IEEE 802.15.4 or Bluetooth IEEE 802.15.1, using directional antennas or triangulation. Finally, the communication technology and the computation required by the protocol must be quick enough to communicate a *Chp* and a *V*s without stopping.

Whatever **interaction** with the *D* should be easy and agile. The *electronic payment system* required in the system should be anonymous and untraceable.

4.2.3 Adversary Model

Interests of *D*s and *SP* could be conflicting. On the one hand, *D*s wish to save money, sometimes in a dishonest way and taking action against the system. On the other hand, *SP* could compromise *D*s' privacy because, in case of fraud, knowing the identity could be helpful. In addition, *SP*, who is desiring to earn more money, could take dishonest actions against *V*s,

by falsely accusing them of fraud. As a result, fraud control and privacy protection could become opposed objectives.

4.3 General Overview

The following section introduces a description of the overall scheme of the *LEZ* system as follows:

The system entities (4.4.1: Setup and 4.4.2: Certification) are initialized before starting the system. Further, *SP* prices the *LEZ* (4.5.1: Price generation), per time unit and emission category, and sends each *Chp* a list of prices signed by the competent entity. *SP* repeats these operations every time prices are updated.

SE generates different credentials for *V* (4.4.3: Certificate generation) every time it enters a *LEZ* (i.e. when the vehicle's engine is started up or just when the *D* is aware of her intentions of entering a *LEZ*) in order to correctly authenticate with *Chp*s and avoid linkability between their itineraries.

A vehicle *V* entering a *LEZ* (4.4.4: Check-in) communicates with a *Chp* and they authenticate each other. Only when the authentication with *V* fails, *Chp* takes a picture of the *V* number plate as evidence of the infringement with which a *proof-of-entrance-incidence* ζ_i is generated. ζ_i is sent to *PA* so as to verify the existence of fraud, and in order to apply the corresponding sanction. If the authentication is correct, *V* gets a *proof-of-entrance* γ_i that includes the entry time.

When a *V* exits the *LEZ* (4.4.5: Check-out), it communicates with a *Chp* and they authenticate each other. In the case of a successful authentication with *V*,*Chp* informs *V* about the exit time and the destination account for payment through a *proof-of-departure* γ_o, which is presented as a receipt. In the case of an authentication failure, *Chp* takes a picture of *V* as a *proof-of-departure-incidence* ζ_o, and it is sent to *PA*.

D computes the amount to pay using γ_i and γ_o, within a specific period of time after the departure. The amount to pay will be set depending on the length of the stay and its emission category. Then, the *D* makes a transaction through an electronic payment system, and a *proof-of-payment* γ_p is sent to *Chp*.

In the figure 4.1, the graphic of *Check-in* and *Check-out* steps are showed.

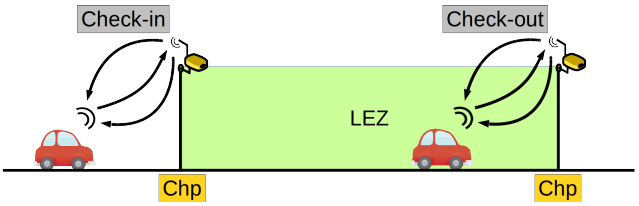


Figure 4.1: Entrance/Departure scheme

The payment verification (4.5.3: Payment verification) is done by *SP* every certain period of time (i.e. monthly). *SP* uses each pair of γ_i and γ_o associated to a same V, to compute the amount of money that V should have correctly paid. Therefore, it checks whether the transaction corresponds with the calculated amount. If that verification is not successful, *SP* sends *PA* a *proof-of-payment-incidence* ζ_p, which contains both γ_i and γ_o.

Finally, *PA* corroborates every ζ received (4.6: Sanction). *PA* only reveals the identity of the V owner (anonymity is revoked) when fraud is verified. Therefore, *PA* requires V to send evidences to refute the accusation. *PA* decides whether to sanction the owner or not after evaluating these evidences.

The previous scheme is composed of *Entrance/Departure LEZ*, *Payment* and *Sanction* protocols. These three protocols, which are related to each other, are detailed in sections 4.4, 4.5 and 4.6. In the tables 4.3 and 4.3, the notation used and the proofs generated, respectively, in these protocols are presented.

4.4 Entrance/Departure LEZ Protocol

In this section, the main phases of the Entrance/Departure protocol, in which the foundations for the main system lay, are detailed in the first five subsections. The two next subsections introduce two different improvements of the main system and address an approach to multiple zones and residents living inside the *LEZ*.

Notation	Name	Description
(Pk_e, Sk_e)	Asymmetric key pair, public and private, belonging to the entity e	
$Enc_e(m)$	Encryption of the message m using the asymmetric public key Pk_e	$Enc_e(m) = E_{Pk_e}(m) = m'$
$Dec_e(m')$	Decryption of the encrypted message m' using the asymmetric private key Sk_e	$E_{Sk_e}(m') = E_{Sk_e}(E_{Pk_e}(m)) = m$
m^*	Set of infomation composed of the message and its signature	$m^* = (m, \bar{m})$
$h(m)$	Hash image of the message m	
$Sign_e(m)$	Digital signature of the message m by the entity e	$Sign_e(m) = E_{Sk_e}(h(m)) = \bar{m}$
$Verif_e(m, \bar{m})$	Verification of the digital signature $\bar{m}$ by the entity e	$Verif_e(m, \bar{m}) = E_{Pk_e}(\bar{m}) = E_{Pk_e}(E_{Sk_e}(h(m))) = h(m)$?
$cert_e$	Public key certificate of the Pk_e of the entity e supplied by CA_i	It includes Pk_e, $sign_{CA_i}(h(Pk_e\|...))$, etc.
$fing_e$	Fingerprint of the public key certificate Pk_e of the entity e	It is computed by using a Hash function of the public key certificate
$cert_e.ext$	Certificate extension ext of the public key certificate $cert_e$	
N_x	Nonce X	It is a random number

Table 4.2: Notation

4.4.1 Setup

The setup process works as follows:

1. *PA*:

   - *PA*, which is a collegiate entity, is constituted by a set of m members
   - An asymmetric key pair (Pk_{PA}, Sk_{PA}) is securely generated by means of a threshold scheme such as in [36]. The Sk_{PA} is divided into m pieces or shares and is securely distributed among the m entities.
   - From a competent authorities (i.e. Police) they obtain its public key certificate $cert_{PA}$, and a certificate repository of the authorities

2. *SP* and *VCA* obtain from competent authorities (i.e. city council and a transit authority, respectively):

   - An asymmetric key pair (Pk_{SP}, Sk_{SP}) and (Pk_{VCA}, Sk_{VCA}), its public key certificate $cert_{SP}$ and $cert_{VCA}$, and a certificate repository of the authorities

   The certificate chain length of *VCA* is 1, and 0 in the case of *SP*. The period of $cert_{SP}$ validity can correspond to the concession lifetime of the service, without exceeding it.

Proofs	Name	Content
θ	*Information-of-prices*, which is sent from *SP* to all the *Chp*s periodically	$\theta = (prices, TS)$, where TS is a timestamp
ψ	*Information-of-entrance*, which is sent from *Chp* to V when it is detected	$\psi = (N_A, \theta^*)$
ω_i	*Authentication-response-of-entrance*, which is sent from V to *Chp* when V receives a ψ	$\omega_i = (\theta^*, N_A, N_B, fing_{Chp})$
wg_i	*Photo-warning-of-entrance*, which is sent from *Chp* to V in case of entrance incidence	$wg_i =$ $(in_o, plt, ts', \theta^*, N_A, N_B, fing_{Chp}, \bar{\omega_i}, cert_{V_q})$
ζ_i	*Proof-of-entrance-incidence*, which is sent from *Chp* to *SP* in case of entrance incidence	$\zeta_i =$ $(in_o, plt, ph, ts', \theta^*, N_A, N_B, fing_{Chp}, \bar{\omega_i}, cert_{V_q})$
γ_i	*Proof-of-entrance*, which is sent from V to *Chp* in case of right execution	$\gamma_i = (ts', \theta^*, N_A, N_B, fing_{Chp}, \bar{\omega_i}, fing_{V_q})$
ρ	*Information-of-payment*, which is sent from *Chp* to V when it is detected	$\rho = (ts'', N_C, acc)$
ω_o	*Authentication-response-of-departure*, which is sent from V to *Chp* V receives a ρ	$\omega_o = (ts'', N_C, N_D, fing_{Chp})$
wg_o	*Photo-warning-of-exit*, which is sent from *Chp* to V in case of departure incidence	$wg_o =$ $(in_o, plt, ts'', N_C, N_D, fing_{Chp}, \bar{\omega_o}, cert_{V_q})$
ζ_o	*Proof-of-departure-incidence*, which is sent from *Chp* to *SP* in case of departure incidence	$\zeta_o =$ $(in_o, plt, ph, ts'', N_C, N_D, fing_{Chp}, \bar{\omega_o}, cert_{V_q})$
γ_o	*Proof-of-departure*, which is sent from V to *Chp* in case of right execution	$\gamma_o = (ts'', N_C, N_D, fing_{Chp}, \bar{\omega_o}, fing_{V_q})$
ω_c	*Authentication-response-of-change*, which is sent from V to *Chp* when V receives ψ in a *change-zone*	$\omega_c = (\theta^*, N_A, N_B, fing_{Chp})$
wg_c	*Photo-warning-of-change*, which is sent from *Chp* to V in case of *change-zone* incidence	$wg_c =$ $(in_c, plt, ts', \theta^*, N_A, N_B, fing_{Chp}, \bar{\omega_i}, cert_{V_q})$
ζ_c	*Proof-of-change-incidence*, which is sent from *Chp* to *SP* in case of *change-zone* incidence	$\zeta_c =$ $(in_c, plt, ph, ts', \theta^*, N_A, N_B, fing_{Chp}, \bar{\omega_c}, cert_{V_q})$
γ_c	*Proof-of-change*, which is sent from V to *Chp* in case of right execution of a *change-zone*	$\gamma_c = (\theta^*, N_A, N_B, fing_{Chp}, \bar{\omega_c}, fing_{V_q}, ts')$
ζ_r	*Proof-of-residence-incidence*, which is sent from *SP* to *PA*	$\zeta_r = (in_v, \gamma_i, \gamma_o, LEZ_{id}, padding)$
γ_p	*Proof-of-payment*, which is generated and sent to V and *SP* by the electronic payment system used	γ_p contains a $reference$ $(hash(\gamma_o{}^*))$
ζ_p	*Proof-of-verification-incidence*, which is sent from *SP* to *PA*	ζ_p=$(in_v, \gamma_i, \gamma_o)$

Table 4.3: Proofs and data generated in the execution of the protocol

3. As it is detailed in Section 3.2.1, *VCA* generates and associates a certification authority VCA_{C_i} for each group of users, that is, for each element of the subset $K = \{C_1, C_2, ..., C_{num_K}\}$, consiting of an asymmetric key pair ($Pk_{VCA_{C_i}}$, $Sk_{VCA_{C_i}}$) and CA certificate $cert_{VCA_{C_i}}$

4. Each *Chp* applies the following steps:

   i. Obtain a certificate repository of the authorities and entities

   ii. Generate an asymmetric key pair (Pk_{Chp}, Sk_{Chp})

   iii. Securely obtain a public key certificate $cert_{Chp}$ from *SP* containing an extension $cert_{Chp}.loc$ with its location coordinates

4.4.2 Certification

In this phase, *Vehicle Certification Authority VCA* registers a user in a group of users (an element of the subset *K*) and securely installs the corresponding Certification Authority VCA_{C_i} associated to C_i in the *SE* of each *V*, consisting of $Pk_{VCA_{C_i}}$, $Sk_{VCA_{C_i}}$ and $cert_{VCA_{C_i}}$. This phase is performed from time to time, for example, before purchasing a vehicle and/or passing the regular technical vehicle tests. Check Section 3.2.2 for further details.

4.4.3 Certificate Generation

The *SE* of the *V*, which has a certification entity VCA_{C_i} installed, generates new credentials. The resulting credentials consist of an asymmetric key pair (Pk_{V_q}, Sk_{V_q}) and a public key certificate $cert_{V_q}$ containing an extension $cert_{V_q}.idS$, which is the encryption of the vehicle identifier V_{id} $Enc_{Pk_{PA}}(V_{id})$. This phase is performed every time a *V* is on its way to enter a *LEZ* in order to avoid linkability between its trips. For example, when the vehicle's engine is started up or just when the *D* decides enter a *LEZ*. Check Section 3.2.1 for more information about this phase.

The repetitive execution of this phase can produce many asymmetric key pairs and their public key certificates, which suppose a large amount of data. These data, together with the proofs generated in the protocol (as it will be seen in the following steps), have to be kept in possession of the user with the aim of using them for defense against a fraud accusation. Due to the limited storage capacities of the *SE*, the storage of all information is

not possible in it. For this reason, it is interesting to securely store the data of previous entrances in the *OBU*. This can be achieved by means of key management custody techniques. The NIST supply the documentation "A Framework for Designing Cryptographic Key Management Systems" [38] and "Recommendation for Key management" [39], which provide general guidance and best practices for the management of cryptographic keying material.

4.4.4 Check-in

When a *Chp* placed in the entrance of a *LEZ* detects a V, the following steps are applied:

1. Chp has to:

   i. Generate a nonce N_A and compose a message *information-of-entrance* $\psi = (N_A, \theta^*)$, where θ^* is detailed in the following Section 4.5.1

   ii. Sign ψ: $Sign_{Chp}(\psi) = \bar{\psi}$, and send ψ^* and its $cert_{Chp}$ to V

2. The *SE* of the V, with the help of the *OBU*, has to:

   i. Verify the certificate $cert_{Chp}$ and the signature $\bar{\psi}$: $Verif_{Chp}(N_A, \theta^*, \bar{\psi})$

   ii. Verify the signature $\bar{\theta}$: $Verif_{SP}(prices, TS, \bar{\theta})$ and the location coordinates $cert_{Chp}.loc$ of *Chp*

   iii. Verify the freshness of TS:$|TS\text{-}current\ time| < \delta$, where δ is a fixed time

   iv. Generate a nonce N_B and compute the fingerprint $fing_{Chp}$ of $cert_{Chp}$ (it is computed as the hash function of the certificate and it is used as an identifier)

   v. Compose a message *authentication-response-of-entrance* $\omega_i = (\theta^*, N_A, N_B, fing_{Chp})$

   vi. Sign ω_i: $Sign_{V_q}(\omega_i) = \bar{\omega}_i$, and send N_B, $\bar{\omega}_i$ and its $cert_{V_q}$ to *Chp*

3. *Chp* has to:

i. Obtain a time ts, and verify the certificate $cert_{V_q}$ and the signature $\bar{\omega_i}$: $Verif_{V_q}(\theta^*, N_A, N_B, fing_{Chp}, \bar{\omega_i})$

ii. If one of the verifications fails:

ii.i Generate an incidence number of entrance in_i

ii.ii Take a photograph ph of V and extract the plate number plt

ii.iii Compose *photo-warning-of-entrance* wg_i= (in_i, plt, ts, θ^*, N_A, N_B, $fing_{Chp}$, $\bar{\omega_i}$, $cert_{V_q}$)

ii.iv Sign wg_i: $Sign_{Chp}(wg_i) = \bar{wg_i}$, and send in_i, plt, ts and $\bar{wg_i}$ to the V

ii.v Compose a message *proof-of-entrance-incidence* ζ_i= (in_i, plt, ph, ts, θ^*, N_A, N_B, $fing_{Chp}$, $\bar{\omega_i}$, $cert_{V_q}$)

ii.vi Sign ζ_i: $Sign_{Chp}(\zeta_i) = \bar{\zeta_i}$, and send ${\zeta_i}^*$ to SP

iii. If the verifications performed in 3i) are correct:

iii.i Compute the $fing_{V_q}$ of $cert_{V_q}$ and compose a message *proof-of-entrance* γ_i=(ts, θ^*, N_A, N_B, $fing_{Chp}$, $\bar{\omega_i}$, $fing_{V_q}$)

iii.ii Sign γ_i: $Sign_{Chp}(\gamma_i) = \bar{\gamma_i}$ and send ts and $\bar{\gamma_i}$ to the V

4. The SE of the V, with the help of the OBU, analyses the received message:

i. If the content of the message corresponds to the data sent in step 3(ii)iv:

i.i. Verify the signature $\bar{wg_i}$: $Verif_{Chp}(in_i, plt, ts, \theta^*, N_A, N_B, fing_{Chp}, \bar{\omega_i}, cert_{V_q}, \bar{wg_i})$

i.ii. Verify the freshness of ts:$|ts\text{-}current\ time| < \delta'$, where δ' is a fixed time

i.iii. Store ${wg_i}^*$

ii. If the content of the message corresponds to the data sent in step 3(iii)ii:

ii.i. Verify the signature $\bar{\gamma_i}$: $Verif_{Chp}(\theta^*, N_A, N_B, fing_{Chp}, \bar{\omega_i}, fing_{V_q}, ts, \bar{\gamma_i})$

ii.ii. Verify the freshness of ts:$|ts\text{-}current\ time| < \delta'$, where δ' is a fixed time

ii.iii. Store γ_i^*

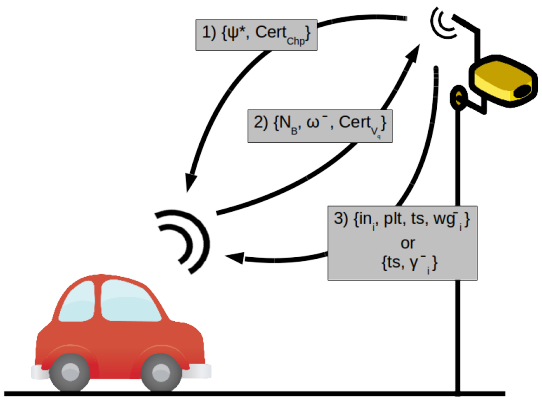


Figure 4.2: Check-in communication

The figure 4.2 shows the messages exchanged between V and *Chp* in the *Check-in* phase.

4.4.5 Check-out

When a *Chp* placed in the exit of a *LEZ* detects a V, the following steps are applied:

1. *Chp* has to:
   i. Obtain a time ts' and a nonce N_C, and compose a message *information-of-payment* $\rho = (ts', N_C, acc)$, where *acc* identifies the target account, of the electronic payment system assumed, of *SP*
   ii. Sign ρ: $Sign_{Chp}(\rho) = \bar{\rho}$, and send ρ^* and its $cert_{Chp}$ to V
2. The *SE* of the V, with the help of the *OBU*, has to:
   i. Verify the certificate $cert_{Chp}$, the signature $\bar{\rho}$: $Verif_{Chp}(ts', N_C, acc, \bar{\rho})$, the location coordinates $cert_{Chp}.loc$ of *Chp* and the freshness of ts':$|ts' - current\ time| < \delta'$, where δ' is a fixed time
   ii. Generate a nonce N_D and compute the fingerprint $fing_{Chp}$ of $cert_{Chp}$
   iii. Compose a message *authentication-response-of-departure* $\omega_o = (ts', N_C, N_D, fing_{Chp})$
   iv. Sign ω_o: $Sign_{V_q}(\omega_o) = \bar{\omega_o}$, and send N_D, $\bar{\omega_o}$ and its $cert_{V_q}$ to *Chp*

3. *Chp* has to:

   i. Verify the certificate $cert_{V_q}$ and the signature $\bar{\omega_o}$: $Verif_{V_q}(ts', N_C, N_D, fing_{Chp}, \bar{\omega_o})$

   ii. If one of the verifications fails:

      ii.i Generate an incidence number of departure in_o

      ii.ii Take a photograph *ph* of the V and extract the number plate plt

      ii.iii Compose a message *photo-warning-of-exit* wg_o= (in_o, plt, ts', N_C, N_D, $fing_{Chp}$, $\bar{\omega_o}$, $cert_{V_q}$)

      ii.iv Sign wg_o: $Sign_{Chp}(wg_o) = \bar{wg_o}$, and send in_o, plt, and $\bar{wg_o}$ to the V

      ii.v Compose a message *proof-of-departure-incidence* ζ_o=(in_o, plt, ph, ts', N_C, N_D, $fing_{Chp}$, $\bar{\omega_o}$, $cert_{V_q}$)

      ii.vi Sign ζ_o: $Sign_{Chp}(\zeta_o) = \bar{\zeta_o}$, and send ${\zeta_o}^*$ to *SP*

   iii. If the verifications performed in 3i) are correct:

      iii.i Compute the $fing_{V_q}$ of $cert_{V_q}$, and compose a message *proof-of-departure* γ_o = (ts', N_C, N_D, $fing_{Chp}$, $\bar{\omega_o}$, $fing_{V_q}$)

      iii.ii Sign γ_o, $Sign_{Chp}(\gamma_o)$=$\bar{\gamma_o}$, and send it to the V

4. The *SE* of the V, with the help of the *OBU*, analyses the received message:

   i. If the content of the message corresponds to the data sent in step 3(ii)iv:

      i.i. Verify the signature $\bar{wg_o}$: $Verif_{Chp}(in_o, plt, ts', N_C, N_D, fing_{Chp}, \bar{\omega_o}, cert_{V_q}, \bar{wg_o})$

      i.ii. Store ${wg_o}^*$

   ii. If the content of the message corresponds to the data sent in step 3(iii)ii:

      ii.i. Verify the signature $\bar{\gamma_o}$: $Verif_{Chp}(ts', N_C, N_D, fing_{Chp}, \bar{\omega_o}, fing_{V_q}, \bar{\gamma_o})$

      ii.ii. Store γ_o^*

The figure 4.3 shows the messages exchanged between V and *Chp* in the *Check-out* phase.

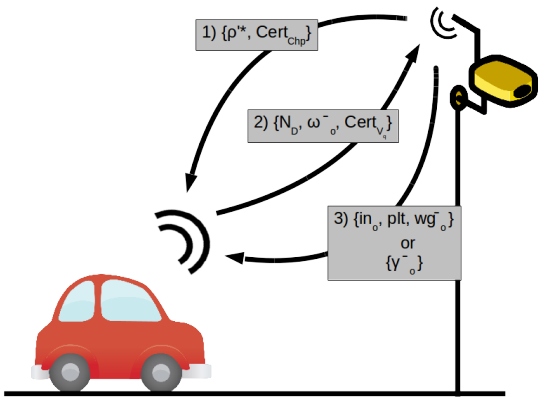


Figure 4.3: Check-out communication

4.4.6 Extending the Protocol: Multiple Zones

In some cases, a *LEZ* could be designed, with a set of adjacent or nested *sub-zones* LEZ_z, where $z = [1, 2, ..., n]$ (according to a set of factors such as the traffic congestion or the contamination level). Each LEZ_z can be associated with different prices according the traffic restrictions[1].

The main idea of the system presented above could be approached in the same way to achieve a fair *multiple LEZ* keeping a high level of user privacy. When a V moves from one zone/sub-zone to another one, it can be understood as a change of *zone*. In order to control this change, new *Chp*s are placed just in the border of the LEZ_z, specifically, in the streets that interconnect booth zones. When a V goes through one of these *Chp*s, the *Chp* records both the exit from the old zone and the entrance to the new zone, in a same *proof-of-change* γ_c. Therefore, several signed proofs γ_c^*, $\gamma_c^{*\prime}$, $\gamma_c^{*\prime\prime}$, etc., as many as change of zones exist, are generated together with γ_i^* and γ_o^*. An example of multiple LEZ composed of two zones, which are nested in each other, can be observed in figure 4.4.

The following steps, which manage the *change-zone* of a V by *Chp*s, are strongly based on the *check-in* phase (see Section 4.4.4 for further details). Therefore, when a *Chp* placed in a *change-zone* detects a V, the following steps are applied:

1. *Chp* composes a message *information-of-entrance* $\psi = (N_A, \theta^*)$, signs ψ

[1] `http://urbanaccessregulations.eu/`

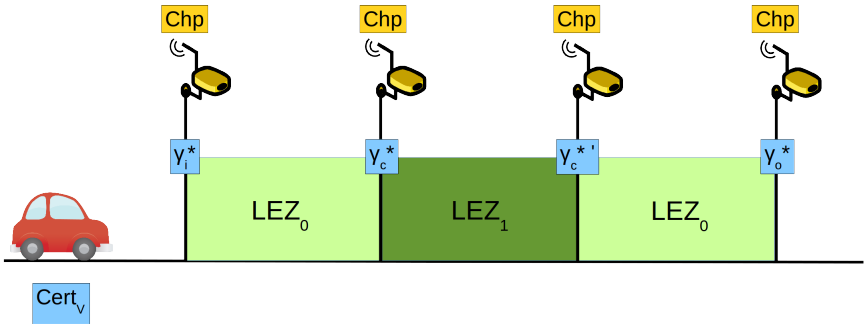


Figure 4.4: Example of multiple LEZ composed of 2 zones

and sends (ψ, $\bar{\psi}$, $cert_{Chp}$) to the V. Note that the *information-of-entrance* proof ψ refers to the entrance to the new zone. In the same way, the prices θ^* contained in ψ belong to the new *zone*

2. the *SE* of the V, with the help of the *OBU*, has to:

   i. Verify $cert_{Chp}$, $\bar{\psi}$, $\bar{\theta}$, $cert_{Chp}.loc$ and ts

   ii. Compose a message *authentication-response-of-change* ω_c = ($\theta^*, N_A, N_B, fing_{Chp}$), sign ω_c, and send (N_B, $\bar{\omega_c}$, $cert_{V_q}$) to *Chp*

3. *Chp* has to:

   i. Verify $cert_{V_q}$ and $\bar{\omega_c}$

   ii. If one of the verifications fails, *Chp* performs the following operations:

      ii.i Take a photograph ph of V and extract the plate number plt

      ii.ii Compose a message *photo-warning-of-change* wg_c=(in_c, plt, ts, θ^*, N_A, N_B, $fing_{Chp}$, $\bar{\omega_i}$, $cert_{V_q}$), sign wg_c, and send (in_c, plt, ts, $\bar{wg_c}$) to V

      ii.iii Compose a message *proof-of-incidence-change* ζ_c= (in_c, plt, ph, ts, θ^*, N_A, N_B, $fing_{Chp}$, $\bar{\omega_c}$, $cert_{V_q}$), sign ζ_c, and send $\zeta_c{}^*$ to *SP*

   iii. If the verifications performed are correct, *Chp* has to:

      iii.i Compose a message *proof-of-change* γ_c=(θ^*, N_A, N_B, $fing_{Chp}$, $\bar{\omega_c}$, $fing_{V_q}$, ts), sign γ_c, and send (ts, $\bar{\gamma_c}$) to the V

4. The *SE* of the *V*, with the help of the *OBU*, analyses the received message:

   i. If the content of the message corresponds to the data sent in step 3(ii)ii, it verifies $w\bar{g}_c$ and ts, and stores $wg_i{}^* = (w\bar{g}_i, w\bar{g}_i)$

   ii. If the content of the message corresponds to the data sent in step 3(iii)i, it verifies $\bar{\gamma}_c$ and the freshness of ts, and stores γ_c^*

4.4.7 Extending the Protocol: Addressing Residents

People or local residents living in a *LEZ* could be discriminated by the application of a *LEZ* in their streets. These people should not have to pay to get to their homes as much as non-residents driving through a *LEZ*. For this reason, the use of a *LEZ* that does not discriminate residents would be an interesting idea.

This preferential treatment to residents is not new. In other contexts such as *disc parking*[2] or *blue zones*, local people can park for free or pay less, for example, with the use of resident cards. In this way, people have to accredit their vehicles in order to make use of the *LEZ* with a preferential treatment. To do that, residents have to prove that their domicile and the fiscal domicile of their vehicles are the same and belongs to the *LEZ*.

The *Entrance/Departure LEZ protocol* proposed above can be improved by addressing resident *V*s. First of all, when a *V* is certified (Section 4.4.2), it is initialized with a *LEZ* identifier LEZ_{id}, which belongs to the *LEZ* where it is accredited. In case *V* is not considered resident, the protocol keeps unchanged.

Subsequently, an extension $Cert_{V_q}.res$ to the $Cert_{V_q}$ is added in step 2 of Section 3.2.3, where the *SE* of a *V* generates a public key certificate $Cert_{V_q}$ every time the *V* is on its way to enter the *LEZ*. This extension contains the probabilistic encryption by using OAEP *padding* of the *LEZ* identifier LEZ_{id} with the public key of *SP*: $Cert_{V_q}.res = Enc_{Pk_{SP}}(LEZ_{id}, padding)$.

After a *V* has been left the *LEZ*, the *SP* can verify whether a *V* is considered resident of its own zone because the $Cert_{V_q}$ used in the *check-in* and *check-out* phase will include $Cert_{V_q}.res$. In the case of a *V* resident considered

[2] `http://www.dublincity.ie/main-menu-services-roads-and-traffic-parking-dublin/parking-city-residents`

in a *LEZ*, a *SP* could verify it only if the *SP* is the one who offers the service of the concerned *LEZ*. Then, the *SP* will be able to check it by decrypting $Cert_{V_q}.res$ with the use of its private key: $Dec_{Sk_{SP}}(Cert_{V_q}) = (LEZ_{id}, padding)$.

In a multiple LEZ, the *LEZ* identifier LEZ_{id} could be replaced by a LEZ_z identifier. In this way, a V considered a resident will have a preferential treatment if in its trajectory it goes through its LEZ_z. In this case, the *SP* should verify that at least one of the proofs γ_i^*, γ_c^*, ..., γ_o^* is associated to the LEZ_{id}.

4.5 Payment Protocol

This section describes the part of the system related to the toll payment. This protocol requires the steps of the previous section, which lays the foundations of the system. The payment protocol is described in the first three subsections: *price generation*, *payment* and *payment verification steps*. Two different improvements are introduced in the two last subsections by modifying the payment protocol (Multiple zones and residents).

4.5.1 Price Generation

Whenever *SP* wants to modify the fees of a *LEZ*, it performs the following operations:

1. Set the *prices* per unit of time and emission category (i.e. European Emission Standards) and obtain a timestamp TS
2. Compose a message *information-of-prices* $\theta = (prices, TS)$
3. Sign θ ($Sign_{SP}(\theta) = \bar{\theta}$) and send θ^* to each *Chp*

4.5.2 Payment

After a V leaves the *LEZ*, regardless of the electronic payment system used, the payment is made by performing the following operations:

1. Recover ts and ts' from a pair of γ_i, γ_o
2. Compute the length of stay τ a *LEZ*: (ts'-ts)=τ

3. Recover the *prices* included in θ^* of γ_i, and compute the *amount* of money required to pay according to τ, its pollutant emissions and the *prices*

4. Compute the *reference*, which will link the transaction and a trip in *LEZ*, as the hash function of $\gamma_o{}^*$: $reference = hash(\gamma_o{}^*)$

5. Make a payment of the *amount* of money, which contains the *reference* in the transfer subject, to the target account *acc*. As a result, a *proof-of-payment* γ_p containing the *reference* is obtained from the electronic payment system used.

The payment of a *LEZ trip* can be made by anyone. The payer could be the same *D*, a passenger or even a business. To do that, the payment proof, regardless of the electronic payment system used, have to include a *reference*. Therefore, the payer, in order to be able to perform the operations detailed above, should know the set of γ_i, γ_o proofs and their $cert_{V_q}$ associated. A good way to securely share this information between owners and payers could be by using an outsourced data protection system in the Cloud with access control based on roles such as in [40].

4.5.3 Payment Verification

Each *Chp* periodically (i.e. every day) sends *SP* the different proofs γ_i, γ_o and ζs (ζ_i and ζ_o), which are generated in phases 4.4.4 and 4.4.5 of the protocol. *SP* stores all these information in a data base and then, forwards the incidences ζ_i and ζ_o to *PA*. Moreover, *SP* obtains a set γ_p from electronic payment system used. Then, *SP* performs the following operations every *payment verification period* (in batch), for example every month, for each set of proofs γ_i and γ_o associated to the same $fing_{V_q}$:

1. Extract *ts*, *ts*′, and $cert_{V_q}.em$ from γ_i and γ_o

2. Extract *prices* from θ^*, included in γ_i

3. Compute the length of stay τ a *LEZ*: (*ts*′-*ts*)=τ

4. Compute the *amount*′ of money required to pay according to τ, $cert_{V_q}.em$ and the *prices*

5. Compute the $reference'$ as the hash function of ${\gamma_o}^*$: $hash({\gamma_o}^*)$

6. Search and verify whether there is a single *proof-of-payment* γ_p with the same reference ($reference' = reference$).

7. Verify whether the transfer was successful and recover the *amount* of money paid

8. Verify whether $amount = amount'$

9. If one of the verifications fails then,

   i. Generate an incidence number of verification in_v

   ii. Compose a message *proof-of-verification incidence* ζ_p including γ_i and γ_o of V_q: ζ_p=(in_v, γ_i, γ_o, γ_p)

   iii. Sign ζ_p ($Sign_{SP}(\zeta_p) = \bar{\zeta_p}$) and send ${\zeta_p}^*$ to *PA*

4.5.4 Extending the Protocol: Multiple Zones

As it was stated in the previous section, when a V moves from a zone/sub-zone to another LEZ_z, it can be understood as a *change* of zone. In each *change-zone*, a time is generated (ts'', ts''', ts'''', etc., which are included in each *proof-of-change* γ_c^*, $\gamma_c^{*\prime}$, $\gamma_c^{*\prime\prime}$, etc.) for each stretch.

After the V leaves definitively a *Multiple LEZ*, a set of γ_i, γ_c^*, $\gamma_c^{*\prime}$, $\gamma_c^{*\prime\prime}$, etc. and γ_o associated to the same V by its $fing_{V_q}$ are generated.

Therefore, the payment phase, which is based on the *single LEZ* payment described above (Sec. 4.5.2) where anyone can pay for a trip in the *multiple LEZ*), is modified taking into account the proofs previously mentioned (like in Section 4.5.2):

1. Compute the length of stay for each stretch *sub-zone*: $\tau = ts' - ts$, $\tau'' = ts'' - ts'$, $\tau''' = ts''' - ts''$, etc.

2. Compute and accumulate the *amount* of money required to pay in each stretch according to the several τ, its pollutant emissions and the *prices*, which are fixed each time the V enters or changes a zone.

3. Compute the *reference* as the hash function of ${\gamma_o}^*$: $hash({\gamma_o}^*)$

4. Make a payment of the *amount* of money, which contains the *reference* in the transfer subject, to the target account *acc*.

In the same way, the payment verification phase, which is based on the previous phase of payment verification (See Sec. 4.5.3 for more details), is modified. Therefore, the *SP* performs the following operations:

1. Extract ts, ts', ts'', ts''', etc. from γ_i,γ_c^*, $\gamma_c^{*\prime}$, $\gamma_c^{*\prime\prime}$, etc. and γ_o

2. Extract the different *prices* from θ^*, $\theta^{*\prime}$, $\theta^{*\prime\prime}$, etc.

3. Compute the length of stay for each stretch *sub-zone*: $\tau = ts' - ts$, $\tau'' = ts'' - ts'$, $\tau''' = ts''' - ts''$, etc.

4. Compute and accumulate the *amount* of money required to pay in each stretch, according to the several τ, its pollutant emissions and the *prices*, which are fixed each time the V enters or changes a zone

5. Compute the $reference'$

6. Search and verify whether there is a single *proof-of-payment* γ_p with the same reference ($reference' = reference$)

7. Verify whether the transfer was successful and whether $amount = amount'$

8. If one of the verifications fails, it composes a message *proof-of-verification-incidence* ζ_p = (in_v, γ_i,γ_c^*, $\gamma_c^{*\prime}$, $\gamma_c^{*\prime\prime}$, ..., γ_o, γ_p), signs ζ_p and sends ${\zeta_p}^*$ to *PA*

4.5.5 Extending the Protocol: Addressing Residents

The protocol, detailed in Section 4.4.7, already provides a vehicle qualification as resident and non-resident condition, and its detection. The system, in order to give a preferential treatment to residents by means of a special economic approach, has to provide a payment protocol applying a *discount* on the *prices* of the *LEZ*.

In such way, a resident V has to apply the *discount* to the *amount* value (at the time of the calculation of the payment and after step 4.5-3)): *discount*:

$amount = amount - (amount * discount)$; and then, make the transfer of the new *amount*.

Since the *SP* is able to detect whether a *V* is considered resident of a *LEZ* (detailed in Section 4.4.7), four cases can take place:

- *SP* detects a *V* considered resident and verifies that the fees are correctly paid as a resident.
- *SP* detects a *V* considered non-resident and verifies that the fees are correctly paid as a non-resident.
- *SP* detects a *V* considered resident and verifies that the fees are incorrectly paid as a non-resident.
- *SP* detects a *V* considered non-resident and verifies that the fees are incorrectly paid as a resident.

In the case of a resident detection, the *SP* verifies whether vehicles have paid correctly according to their resident condition, after step 4.5-5), applying the *discount* to $amount'$ value in a similar way as the above mentioned: $amount' = amount' - (amount' * discount)$.

If the payment is incorrectly paid as a resident when it is not a resident, just as in the last case, the *SP* has to:

1. Generate an incidence number of verification in_v
2. Compose a message *proof-of-residence-incidence* ζ_r including γ_i, γ_o, *padding* and LEZ_{id} of the V_q: ζ_r=(in_v, γ_i, γ_o, LEZ_{id}, *padding*).
3. Sign ζ_r ($Sign_{SP}(\zeta_r) = \bar{\zeta_r}$) and send ${\zeta_r}^*$ to *PA*

4.6 Sanction Protocol

This section makes the entire system take shape in combination with the other two protocols. The first subsection describes the sanction process. The two next subsections make reference to the modifications of the sanction process to approach the system to multiple zones and resident extensions.

4.6.1 Sanction

PA, as the entity responsible for auditing and penalizing irregularities in the execution of the protocol and potential frauds, receives a set of incidences proofs from *SP* every sanction period (i.e. 15 days from the *payment verification period*). Depending on the type of proof, the sanction process is carried out as follows:

1. In the case of ζ_i or ζ_o:
    i. The m sub-entities of *PA* perform the following operations:
        i.i. Verify the signature ζ^* and the signatures included in ζ, and extract the number plate plt from the photograph ph, included in ζ
        i.ii. Contact the owner of the V using plt, inform her about the sanctioning procedure and require evidences of the contrary to refute the accusation
2. In the case of ζ_p:
    i. The m sub-entities of *PA* perform the following operations:
        i.i. Verify all the signatures included in ζ_p and verify that the signatory of γ_i and γ_o is the same
        i.ii. Verify the right payment by repeating steps 1-8 from 4.5.3
        i.iii. If the incidence is confirmed by a least t from m members of *PA*, the $fing_{V_q}$ of $cert_{V_q}$ is then made public, for example, in a *Bulletin Board*, which is widely used in many countries such as the "Tablón Edictal de Sanciones de Tráfico" of the Spanish Ministry of Internal Affairs[3]. In a private zone of the *Bulletin Board*, the proofs presented by the *SP* are uploaded
    ii. The owner of the V performs the following operations:
        ii.i. Look for fingerprints $fing_{V_q}$ of her $Cert_{V_q}$ in the *Bulletin Board* every certain period of time

[3] `https://sede.dgt.gob.es/es/tramites-y-multas/alguna-multa/consulta-tablon-edictal-testra/`

ii.ii. If $fing_{V_q}$ is not found, all the information generated by system and stored in the *OBU*, which are associated to the same $Cert_{V_q}$, are deleted

ii.iii. If $fing_{V_q}$ is found, then:

ii.iii.i. Obtain more information of the sanction process, through the authentication of her $cert_{V_q}$, if one of her fingerprints is found on the Bulletin Board, and get access to the private zone of the *Bulletin Board*

ii.iii.ii. After evaluating the proofs of her accusation, she could present evidences to *PA* in order to refute her accusation

iii. The m sub-entities of *PA* have to:

iii.i. Verify the evidences presented by the owner of the V to refute the accusation

iii.ii. The Sk_{PA} is reconstructed from their own shares if the presented contra evidences are not considered valid by at least t from m members of *PA*. Therefore, the identifier V_{id} of V_q is recovered by opening the extension $certV_q.idS$ of the certificate $certV_q$, included in the proof γ_i or γ_o. Finally, the owner of V is fined according to the type of infraction

4.6.2 Extending the Protocol: Multiple Zones

PA receives a set of incidences proofs from *SP* as in the previous subsection. Apart from that, considering a *multiple LEZ*, *PA* can also receive *proof-of-incidence change* ζ_c. In spite of this difference, the sanction process is similar to the process described above. Depending on the type of incidence, the sanction process is carried out as follows:

1. In the case of ζ_i, ζ_o or ζ_c, the m sub-entities of *PA* verify ζ^* and the included signatures, recover *plt* from *ph*, and request evidences by the owner of the V to refute the sanctioning procedure

2. In the case of ζ_p:

   i. The m sub-entities of *PA* performs the following operations:

i.i. Verify all the signatures included in ζ_p and verify that the signatory of γ_i,γ_c^*, $\gamma_c^{*\prime}$, $\gamma_c^{*\prime\prime}$, etc. and γ_o, is the same

i.ii. Verify the right payment by repeating steps 1-7 from 4.5.4

i.iii. If the incidence is confirmed by a least t from m members of *PA*, the $fing_{V_q}$ of $cert_{V_q}$ is then made public. In a private zone of the *Bulletin Board*, the proofs presented by the *SP* are uploaded

ii. The owner of the V looks for her $Cert_{V_q}$ in the *Bulletin Board*. If she finds it, she then gets more information through the private zone of the *Bulletin Board*. After evaluating the proofs of her accusation, she could present evidences to refute *PA*'s accusation

iii. The m sub-entities of *PA* verify the evidences presented by V. If they are not considered valid by at least t from m members of *PA*, the Sk_{PA} is reconstructed. Therefore, the identifier V_{id} is recovered. Finally, the owner of V is fined

4.6.3 Extending the Protocol: Addressing Residents

The *PA* can receives incidences ζ_r, which are related to a misuse of resident condition, from *SP*. In this case, the ζ_r incidence proof is dealt by *PA* as a *proof-of-verification incidence* ζ_p and thus, it is processed in step 4.6.1-2). Due to this protocol extension, the basic protocol described in 4.6.1 has to be modified. In particular, step 4.6.1-2ii), which are performed by the m sub-entities of *PA*, are changed by the following ones:

i.i. Verify all the signatures included in ζ_p and verify that the signatory of γ_i and γ_o is the same

i.ii. Recover $Cert_{V_q}.res$ from $Cert_{V_q}$

i.iii. Compute $Cert_{V_q}.res'$ using LEZ_{id} and $padding$ included in ζ_r: $Cert_{V_q}.res' = Enc_{Pk_{SP}}(LEZ_{id}, padding)$

i.iv. Verify $Cert_{V_q}.res = Cert_{V_q}.res'$

i.v. Verify the right payment by repeating steps 1 to 8 from 4.5.3. Note that a discount has to be applied to the amount when the resident condition is confirmed, such as in Section 4.5.5

i.vi. If the incidence is confirmed by a least t from m members of *PA*, the $fing_{V_q}$ of $cert_{V_q}$ is then made public, for example in a *Bulletin Board*, which is widely used in many countries. In a private zone of the *Bulletin Board*, the proofs presented by the *SP* are uploaded

The remaining steps of the basic protocol keep unaltered.

4.7 Security and Privacy Analysis

The security and privacy requirements of this system are studied in this section. The discussion is organized in three propositions, and each proposition can have several claims to support its fulfillment.

Proposition 4.7.0.1 *The proposed system preserves authenticity, non-repudiation and integrity for the entrance and departure proofs.*

Claim 1. *The creation of fraudulent entrance and departure proofs is computationally unfeasible nowadays.*

Proof. Entrance proofs have the following form γ_i=(ts, θ^*, N_A, N_B, $fing_{Chp}$, $\bar{\omega_i}$, $fing_{V_q}$). The checkpoint signs entrance proofs γ_i: $Sign_{Chp}(\gamma_i) = \bar{\gamma_i}$ and sends the pair ts and $\bar{\gamma_i}$ to vehicles. In the same way, departure proofs γ_o = (ts', N_C, N_D, $fing_{Chp}$, $\bar{\omega_o}$, $fing_{V_q}$) are signed by the checkpoint, $Sign_{Chp}(\gamma_o)$=$\bar{\gamma_o}$, and sent it to vehicles. For these reasons, the generation of entrance and departure proofs is nowadays computationally unfeasible, without the knowledge of the secret key used by *Chp* in the signature. □

Claim 2. Chp*s, which are issuers of the entrance and departure proofs, can not deny the emission of these proofs.*

Proof. Entrance and departure proofs are generated and signed by their issuer (*Chp*s) and, considering the signature scheme secure, this operation can be only performed by these issuers. Thus, the issuer's identity is linked to the proofs and, for the properties of the electronic signature scheme, it can not deny its authorship. □

Claim 3. *The content of the entrance and departure proofs can not be modified by the vehicles.*

Proof. If we suppose that the signature scheme is secure and that the hash summary function is collision-resistant, if the content of the entrance or departure proofs was modified, the verification of the signature would be incorrect because $Sign_e(m) = E_{Sk_e}(h(m)) = \bar{m}$. In order to pass the verification, the signature would be regenerated from the new entrance or exit ticket. This operation is computationally unfeasible without the knowledge of the checkpoint secret key. □

Result 1 *According to the presented proofs in Claims 1, 2 and 3, it can be assured that the protocol satisfies the needed security requirements (authenticity, integrity and non-repudiation) for the proofs to be considered valid.*

Proposition 4.7.0.2 *The system presented here preserves the privacy of its users and protects their anonymity, avoiding the traceability of their actions.*

Claim 4. *The system guarantees the anonymity of honest users.*

Proof. The information that the user transmits to enter the system is the *authentication-response-of-entrance* $\omega_i = (\theta^*, N_A, N_B, fing_{Chp})$ and its signature. The Chp will verify the signature using the certificate $cert_{V_q}$ accompanying the user message. This certificate (generated by the SE of the V before entering the LEZ) identifies the vehicle, but the identification information is protected with an asymmetric encryption, using the public key of PA. Thus, Chp can verify the signature but it is not able to identify the vehicle. Then, Chp generates and transmits to the user $\bar{\gamma}_i$. With this evidence, the vehicle may enter the LEZ. The information related to the user identification inside $\bar{\gamma}_i$ is the same included in ω_i. This means that the V enters the LEZ without being identified.

When V leaves the LEZ, the user must send the *authentication-response-of-departure* $\omega_o = (ts', N_C, N_D, fing_{Chp})$ to the Chp. It is not possible to identify the user through the signature on ω_o by the reasons explained in the previous paragraph. Consequently, the entrance a LEZ and the departure

from a *LEZ* of honest users are anonymous. □

Claim 5. *The protocol does not allow to trace or to link the actions of the vehicles.*

Proof. It is not possible to associate the various entries and departures from a *LEZ* of the same vehicle if we use the information generated by the execution of the protocol. This happens because the steps described in 3.2.3 are executed each time the V approaches a *LEZ*. This means that the *SE* generates a new $cert_{V_q}$ for the vehicle in each new entrance process. This certificate is the only element that could identify the V. However, considering that the use of the certificate is unique for each entrance/departure process, nobody can relate the identity of V of this entrance/departure process with any other entrance/departure process.

The information that could be repeated in other entrance/departure process of the same V is the $certV_q.idS$ field. But, as it is specified in the protocol, $certV_q.idS$ is computed with a probabilistic encryption, using for example the OAEP padding system, which means that the result of each new encryption credentials is different. □

Result 2 *The system presented here preserves the privacy in accordance with claims 4 and 5: users can use the system anonymously and each new usage can not be related to any other with respect to the identity of the vehicles.*

Proposition 4.7.0.3 *The system has anti-fraud requirements concerning the correctness and verifiability of the evidences generated in the protocol.*

Claim 6. *The system can identify dishonest users thanks to the anonymity revocation property of the protocol.*

Proof. If users do not properly perform the authentication at the entrance/departure process of the system, then they can lose the anonymity because the *Chp* takes a picture of the V capturing the number plate. This information is sent to *PA* to act as it is specified in the *Sanction* protocol. In

the execution of this protocol, *PA* has the ability to identify the user through the number plate.

If users have not made the proper payment, then *SP* verifies in the *Payment Verification* phase that the amount paid corresponds to the tax determined for τ and emissions of V. If the verification fails, this information is sent to *PA* to issue a traffic fine for the user. *PA* verifies the incidence and identifies the user by opening the field $certV_q.idS$ of the certificate with its private key. Obtaining V_{id} allows the identification and the punishment of the dishonest user. □

Claim 7. *The protocol execution generates evidences for an honest user (they are saved in the* OBU*) to prove or disprove the allegations of fraud.*

Proof. When a user is accused of not performing the authentication correctly, a ζ_i or a ζ_o that records the incidence is generated. The user can be accused of using an improper certificate $cert_{V_q}$ or sending an incorrect signature $\bar{\omega}_i/\bar{\omega}_o$. In both cases, *PA* contacts her during the *Sanction* process, so she can provide evidences to rebut the charges.

An honest user can retrieve a valid $cert_{V_q}$, from her *OBU*, which matches her vehicle (identified by the number plate) or a signature $\bar{\omega}_i/\bar{\omega}_o$ that was successfully computed by the *SE* with the help of the *OBU* during the entrance/departure process of the *LEZ*.

In case of a payment incidence, the user has to demonstrate that the payment has been made according to the data stored in ${\gamma_o}^*$ and ${\gamma_p}^*$ (both items signed by the *Chp*). Therefore, an honest user will be able to retrieve this information from her *OBU* and send it to *PA* to refute the accusation. □

Result 3 *The system keeps fraud under control and it can identify dishonest users. These users receive the appropriate traffic fine. The protocol also allows honest users to get evidences of their correct performance. The evidences are used to rebut any traffic fine due to some kind of malfunction of the system's actors.*

4.8 Functional Requirements Analysis

This section presents a feasibility study of the application of the proposed system in the real world. This study is divided into two different parts, the *online-feasibility* study and the *Electronic Payment System Adaption* study, according to the steps of the protocol that are evaluated.

The first one, the *online-feasibility* study, aims to evaluate the most critic part of the system and to demonstrate that its execution is possible nowadays. This study involves sections 4.4.3, 4.4.4 and 4.4.5, which have hard temporary restrictions since communication between the entities is made in movement. In particular, *V*s, which keep moving while entering or leaving the *LEZ*, and *Chp*s, have to be able to properly perform each step of these phases with sufficient time to transmit and receive the expected set of proofs and data (such as certificates, *proofs-of-entrance* or *proofs-of-departure*).

The second study, the *Electronic Payment System Adaption* study, attempts to evaluate whether the existing electronic payment systems accomplish the restriction of anonymity and untraceability, which were stated in Section 4.2. This study also considers some current electronic payment systems to be adapted in the proposed system.

4.8.1 Online-feasibility Study

This study assesses the feasibility of the practical deployment of some parts of the system. In particular, it focuses on the parts of the system that have high temporary restrictions since the information transmitted between some entities is in movement. This study thus evaluates whether the execution of those parts can be performed quickly enough to have time to transmit and receive the required information in movement. The parts of the protocol with high temporary restrictions are the *Certificate Generation* phase (Section 4.4.3), the *Check-in* phase (Section 4.4.4) and the *Check-out* phase (Section 4.4.5). In the two last phases, *V*s and *Chp*s execute the protocol and communicate between them while the first ones pass through the second ones at the entrance/departure of a *LEZ*. In particular, *V*s, *OBU*s, *SE*s, and *Chp*s are involved.

The following development environment and hardware have been used in this study:

- *Mobile Security Card SE 1.0*[4]: This is a Java Card in microSD format, which provides security capabilities for smart phones. Specifically, it uses version 5.0 of the Sm@artCafé Expert's operating system. This device has been used as the *SE* in the simulation because it is a cheap tamper-proof device with cryptographic capabilities.

- *Samsung Galaxy S3*: This smart phone, which uses the Android 4.1.2 version, has been used as an *OBU*. This device has a CPU Quad-core 1.4 GHz Cortex-A9 and 1 GB RAM. In addition, it has a microSD card slot, which jointly with Android OS, they make communication between smart phones and Mobile Security Card possible. The library Spongy Castle, which is a library for Android platform containing the last version of the cryptographic library Bouncy Castle, is used to perform the less sensitive cryptographic operations out of the *SE*, such as signature and certificate verifications. This device plays the role of *OBU* in the simulation.

- *PC*: It is a common personal computer with a CPU of 2 i7-cores of 2,3 Ghz and 1 GB RAM. The used OS is a 64 bit Ubuntu 14.04. This device will perform the operation of the *Chp* in the simulation. These operations have been developed using the Java language and the cryptographic operations have been computed using the last version of Bouncy Castle, which is a Java library.

The study consists in obtaining the performance cost of the system execution and in evaluating whether this cost is low enough to make communication in movement possible using an affordable hardware and easier to find in the market.

Concerning the technology used to communicate between *V*s and *Chp*s, ZigBee technology is considered. In particular the cost of this communication has been estimated from [41], which uses an implementation of ZigBee named XBee[5] and which takes into account the Doppler effect on the communication in movement. The results obtained in [41] prove that the communication in movement between a vehicle at 60km/h and a fixed point

[4] `http://www.gi-de.com/gd_media/media/press/prs_1/pdf/PM_MicroSD_Card_SE_10_final_E.pdf`

[5] `http://www.digi.com/lp/xbee/`

with a low number of erroneous packets is possible. In detail, 2000 packets can be correctly received, and even more if the velocity of the vehicle is lower. These 2000 packets can be transmitted from 250m of distance, which means that approximately 133 packets per second can be transmitted along it, and that each packet can be fitted with 100 bytes of data.

In view of the aforementioned situation, the total temporal cost has been calculated by implementing the steps of the protocols and by estimating the communication cost. The computational costs were gathered by averaging the single processes over 100 iterations. The details of the implementation and the performance results obtained in each phase are detailed below:

1. Certificate Generation (Section 3.2.3): The generation of an entity certificate every time a V is on its way to enter a *LEZ* is implemented. It consists in first generating a RSA key pair (Pk_{V_q},Sk_{V_q}) of 2048 bits in the Mobile Security Card, and then the public key certificate $cert_{V_q}$ following the features described in step 2 (such as the extension containing the encrypted V_{id} with the public key of *PA*). All these have been achieved thanks to the installation of VCA_{C_i} certification entity in the Mobile Security Card. The temporal cost of certificate generation in the Mobile Security Card, which follows the features described in the protocol, is 10,350s, and the size of the generated certificate stored in the smart phone memory is 1574 bytes. Note that the temporal cost covers the key pair generation, the certificate generation and the extraction of it outside the *SE*.

2. Check-in (Section 4.4.4): The cost related to the interaction between *V*s and *Chp*s to get access to the *LEZ* is simulated here. The temporal cost of this phase, which is separated according to the computation and communication costs, is detailed below for each step. In addition, the size of the transmitted information is also detailed.

   - Step 1: Total = 0,228s.
     - Computational cost: 0,020s.
     - Communication cost: 0,208s (3186 bytes = 32 data packets).
   - Step 2: Total = 1,782s.
     - Computational cost: 1,542s.

  - Communication cost: 0,240s (3686 bytes = 37 data packets).

- Step 3: According to the results of the verifications performed in this step, some of the following sub-steps will be either performed or not. If the verifications are correctly performed (step 1 and 3), Total = 0,198s. Otherwise (step 1 and 2), Total = 0,384s. The detail of the sub-steps is as follows:
  - Step 3i: 0,011s.
    * Computational cost: 0,011s.
    * Communication cost: None.
  - Step 3ii: Total = 0,373s. Note that the time required to obtain the license plate of the V from the camera is taken from [42].
    * Computational cost: 0,023s.
    * License recognition cost: 0,175s.
    * Communication cost: 0,175s (2603 bytes = 27 data packets).
  - Step 3iii: Total = 0,187s.
    * Computational cost: 0,018s.
    * Communication cost: 0,169s (2582 bytes = 26 data packets).
- Step 4i: Total = 0,005s. Note that this step will be performed if the verifications performed in 3i fail.
  - Computational cost: 0,005s.
  - Communication cost: None.
- Step 4ii: Total = 0,005s. Note that this step will be performed if the verifications performed in 3i are correct.
  - Computational cost: 0,005s.
  - Communication cost: None.

Supposing that the execution of the protocol is performed correctly, when an incidence has not been generated, that is, the verifications performed in 3i are correct, the total temporal cost is 2,213s. On the contrary, if an incidence has been generated, that is, one of the verifications performed in 3i fails, then the temporal cost is 2,400s.

Therefore, both results are fast enough in order for them to have time to communicate in movement. The temporal length of communication window, in which both entities are able to transmit information at 60 km/h along 250m of distance, is 15s. There is therefore a high margin between 2,400s and 15s, and even more if the velocity of the vehicle is lower.

3. Check-out (Section 4.4.5): The cost related to the interaction between *V*s and *Chp*s to leave the *LEZ* is simulated below. As in the previous phase of the protocol, the temporal cost of this phase, considering the computation and communication costs, is detailed for each step jointly with the size of the transmitted information.

   - Step 1: Total = 0,126s.
     - Computational cost: 0,009s.
     - Communication cost: 0,117s (1793 bytes = 18 data packets).
   - Step 2: Total = 1,499s.
     - Computational cost: 1,317s.
     - Communication cost: 0,182s (2768 bytes = 28 data packets).
   - Step 3: According to the results of the verifications performed in this step, some of the following sub-steps will be either performed or not. If the verifications are correctly performed (step 1 and 3), Total = 0,036s. Otherwise (step 1 and 2), Total = 0,255s. The detail of the step is divided into the following sub-steps:
     - Step 3i: 0,004s.
       - Computational cost: 0,004s.
       - Communication cost: None.
     - Step 3ii: Total = 0,251s. The time required to obtain the license plate of the *V* from the camera is taken from [42], like in the previous phase.
       - Computational cost: 0,011s.
       - License recognition cost: 0,175s.
       - Communication cost: 0,065s (929 bytes = 10 data packets).
     - Step 3iii: Total = 0,032s.

    - Computational cost: 0,013s.
    - Communication cost: 0,019s (267 bytes = 3 data packets).

- Step 4i: Total = 0,003s. Note that this step will be performed if the verifications performed in 3i fail.
  - Computational cost: 0,003s.
  - Communication cost: None.
- Step 4ii: Total = 0,004s. Note that this step will be performed if the verifications performed in 3i are correct.
  - Computational cost: 0,004s.
  - Communication cost: None.

Supposing that the execution of this protocol is performed correctly, when an incidence has not been generated, that is, the verifications performed in 3i are correct, the total temporal cost is 1,665s. On the contrary, if an incidence has been generated, that is, one of the verifications performed in 3i fails,, then the temporal cost is 1,883s.

Therefore, both results are fast enough in order for them to have time to communicate in movement. The temporal length of communication window, in which both entities are able to transmit information at 60 km/h along 250m of distance, is 15s. Thus, there is a high margin between 1,883s and 15s, and even more if the velocity of the vehicle is lower.

After these detailed simulations, the online feasibility of the system has been stated. In particular, the capability of generating a certificate in a *SE* accomplishing with the protocol within a reasonable period of time, has been proven. Both parts of the protocol with higher temporal restrictions, *Check-in* and *Check-out*, have been also demonstrated to be feasible in movement.

4.8.2 Study of the Electronic Payment System Adaption

The electronic payment system is used by the users to pay the charged fees associated to the use of the *LEZ* after the *V* leaves the *LEZ*. The payment system, as it is specified in the model section 4.2.2, has to be anonymous and untraceable, that is, the payment has to prevent from the identification of the

payer and from the link of payments made by the same payer. Otherwise, the payment could be used by the *SP* to identify the users of the *LEZ*, and even to know about the vehicles' routes.

The electronic payment system has to be disassociated from the *LEZ* system, so that:

- The payment could not just be made by the *D*, but also by other passengers or even by a business.
- The payment is made a posteriori, after the *V* has left the *LEZ* and within a billing period.
- The users of the *LEZ*, who have not payed or who have an incorrect payment associated to their trips, have to be sanctioned. Thus, the interest of the users will be to get a proper payment.

Under these conditions, there are some suitable electronic payment systems: Bank transfers (i.e. by means of a payment card), traditional electronic money (i.e. eCash [43]) and other electronic coins based on hashes (such as Bitcoin [44]). The first one, which is well known, is based on a Third Trusted Party (TTP) and on the use of a numeric identifier (i.e. an account number or card number), which keep the payer identity protected. The second one, which is anonymous by definition and does not require from a TTP, has been applied minimally in the real world due to some usability limitations. The last one is a recent digital money based on a peer-to-peer system, which was proposed by Satoshi Nakamoto. In this kind of electronic coins, users can transfer money with no intermediaries by means of hash functions and other cryptographic techniques.

The last payment system stands out above the others due to the absence of TTP and its wide real application. Despite of this, it suffers from a lack of privacy due to its public ledger. Bitcoin transactions or blockchains are associated to users by means of pseudonyms and are composed of other identifiable information such as the sender and receiver public address, among other metadata. All these historic information, which by design composes each Bitcoin, is publicly stored permanently and is accessible to everyone. As a result, Bitcoin traceability is possible climbing between transactions. The transactions of a Bitcoin can be followed from the last to its origin,

thus obtaining more details about each transaction. Even more, blockchain analysis techniques could be used to deanonymize part of the Bitcoin transactions, linking several public addresses to a same user. By means of data mining techniques, it can be performed by analyzing the blockchain path of the change of transaction.

Some solutions to this traceability issue, which are more suitable than others, can be found in the literature. From less to more privacy measures, each user could create several public addresses in order that tracing becomes a difficult purpose. However, it does not provide enough privacy since it is impossible to generate infinite addresses. Another measures is to launder Bitcoins through a TTP, which works as a mix network [45], randomizing a set of Bitcoins anonymously. In this way, Bitcoins of different owners are exchanged without knowing their origin and they are received in new addresses, thus breaking the owner thread. Even so, Zerocoin [46] is able to improve it by removing the TTP. This is an extension of the Bitcoin system, which allows the users to purchase Zerocoins and to later recover Bitcoins, using cryptographic accumulators, commitments and Zero-knowledge proofs, avoiding any possible link between the addresses. Therefore, the use of Bitcoins with the Zerocoin extension can be suitable as the electronic payment system of the *LEZ* system, since it accomplishes with the anonymity and untraceability requirements specified above.

In addition, as it has been seen in Section 4.5.2, the use of a *reference* linking between transaction and a trip in *LEZ* (γ_o) is needed. Besides, it helps to solve some attacks related to the payment such as double spending. Consequently, the electronic payment system used will add the *reference* in the transaction.

In bank transfers, it is possible to add a reference to the transfer subject. In other systems, such as Bitcoin, there is a specific setting which allows the sender to add any information of what she wants.

In the case of the Bitcoin system, the sender can add an optional memo in the payment message. Therefore, Bitcoins can be easily adapted considering that the *payment message* generated after a transaction can be used as a *proof-of-payment* γ_p. The *Payment verification* phase will be carried out after the network nodes have performed their verifications of the blockchains. If more quicker verifications are required, nodes could be reward, as it is considered

in the Bitcoin system. In particular, step 6 of Section 4.5 will be performed by nodes of the network. The *SP* will only have to verify that the verifications of the nodes have been properly performed and that the result is correct.

Hence, the existence of electronic payment systems to be adapted in this system is stated, which accomplish the required properties (anonymity and untraceability). These systems could be easily modified to address the payment in the *Check-out* process at the time of departure the *LEZ*. In addition the anonymity and untraceability requirements, the payment should accomplish a hard temporary requirement, in which this systems sould be quick enough to be performed in movement together with the *Check-out* phase.

Chapter 5

Privacy-Preserving Electronic Road Pricing System for Low Emission Zones with Dynamic Pricing

In this chapter, a new Electronic Road Pricing (ERP) is proposed with the aim to detect fraud while preserving drivers' privacy. More concretely, it provides a non-probabilistic fraud control and the control points only take pictures of the vehicles that misbehave. Last but not least, the proposed system applies to an enhanced dynamic pricing that can help the authorities to better distribute traffic over the road network.

The novelty of the system is introduced in §5.1. In §5.2, the basis of system, the architecture, the involved participants, and the requirements are presented. After that, §5.3 gives an overall idea of the system scheme. The main scheme is composed of Setup Protocol (§5.4), Certification Protocol (§5.5, Payment Protocol (§4.5), Entrance Protocol (§5.7), and Off-line Protocol (§5.8). Security and privacy requirements are evaluated in §5.9. Finally, the analysis of functional requirements is presented in §5.10.

Contents

5.1 Novelty of the Approach 66

5.2 System Model 67

5.2.1 Architecture and Participants 67

5.2.2 Requirements 69

5.3 General Overview 71

5.4 Setup Protocol 75

5.5 Certification Protocol 76

5.5.1 Certificate Entity Installation 76
5.5.2 Certificate Generation 76
5.6 Payment Protocol 76
5.6.1 Price Generation 77
5.6.2 Ticket Acquisition 77
5.7 Entrance Protocol 80
5.8 Off-line Protocol 81
5.8.1 Doublespending Verification 81
5.8.2 Sanction Process 82
5.9 Security and Privacy Analysis 85
5.9.1 Adversary Model 85
5.9.2 Security Analysis 87
5.10 Functional Requirements Analysis 96
5.10.1 On-line Feasibility Study 97
5.10.2 Traffic Simulation 102

5.1 Novelty of the Approach

In this chapter, a new *ERP* system for Low Emission Zones (*LEZ*s), which follows a dynamic pricing approach independently for each road stretch, is proposed. Unlike the proposal presented in Chapter 4, this systems allows a distribution of the traffic in the *LEZ* because the toll costs can be updated according to the use of the stretch. The system generates traffic feedback in real time, which is useful for such purpose. Drivers would then be able to re-plan their routes according to the toll costs, and thus, load-balance the traffic.

As it is stated in Chapter 2, the systems that can be found in the literature ([1, 3, 4, 2, 5, 6, 7, 8, 9]) detect fraud probabilistically by taking photos of vehicles at random places. These systems use this information to verify whether drivers are honest when paying for the covered path. Moreover, the set of photos of vehicles taken by *checkpoints* at random places generates a loss of drivers' privacy, since all the vehicles, honest or not, are registered in a set of points. The system proposed provides a non-probabilistic fraud

control while preserving privacy of honest drivers as in the previous proposal presented (Chapter 4). In this way, only vehicles that misbehave are registered and identified by means of anonymity revocation. In particular, honest drivers keep their privacy and their identity, and their routes are not disclosed. For this reason, drivers are expected to collaborate with the system and to behave correctly in order to keep their privacy. Moreover, unlike other systems, *OBUs* do not register vehicles' geo-locations.

The resulting scheme has been tested in terms of deployment feasibility and the results show that the provided proposal is realistic and it may be deployed in practical scenarios. In addition, this works includes a simulation in order to assess the effect of dynamic pricing of stretches in a traffic network.

The new *ERP* system, which is presented in this chapter, is supported by:

- Roger Jardí-Cedó, Jordi Castellà-Roca, Alexandre Viejo "Privacy-Preserving Electronic Toll System with Dynamic Pricing for Low Emission Zones" In *9th International Workshop on Data Privacy Management (DPM 2014)* pp. 327–334. 2014.

- Roger Jardí-Cedó, Jordi Castellà-Roca, Alexandre Viejo "Privacy-Preserving Electronic Road Pricing System for Low Emission Zones with Dynamic Pricing" In *International Journal of Security and Communication Networks* 2015. Under review.

5.2 System Model

In this section, a conceptual model of the system is presented. This model includes definitions of the architecture and the participants, as well as requirements of the system.

5.2.1 Architecture and Participants

Driver D is the person who drives a vehicle in a *LEZ*. *Vehicle V* is the means of transport registered by a unique *D* (the owner of it) but it may be driven by several *D*s. *V* has an identifier (the vehicle plate) that connects it to the owner. Each *V* has a *Secure Element SE* and an *On-board unit OBU*. The first is

a tamper-proof security module installed in each *V* by the competent traffic authority. It performs all sensitive operations to ensure the security requirements; The second is a device with more computational power and storage capacity than *SE*. This device connects *SE* with the user and performs the less sensitive protocol operations. It has location (GPS) and wireless communication capabilities.

A *LEZ* is a restricted area where vehicles can access in exchange for a payment according to the pollutant emissions of the vehicle. A *LEZ* is divided into a set of street stretches. A *stretch* is a one-way section of street where *V*s have to pay every time they drive through it. Each stretch is divided into a *payment area* and a traffic *restricted area*. A *Beacon* is a device, placed at the *payment area*, which constantly warns the *V*s entering the stretch. A *Checkpoint Chp*, placed at the entrance of the *restricted area* of a *stretch*, aims to control the access of vehicles that enter the stretch. *Service Provider SP*, which manages both, offers an ERP service for urban areas thanks to a concession contract with the local public administration (i.e. City Council). This entity, apart from having the right to offer this service, is responsible for managing the system. *SP* is also responsible for pricing each *stretch* according to its traffic density. *Ticket Provider TP*, which has access to the prices and which is also managed by *SP*, issues tickets to *V*s when they are driving on the *payment area*. The proposal also considers other actors maintained by other authorities: *Vehicle Certification Authority VCA*, which is maintained by the traffic competent authority and provides credentials to *V*s; *Payment Service PS* is a trusted entity that depends on the electronic payment system (i.e. Visa or Mastercard). It enables *D*s to pay the circulation fees to *Service Provider*. *PS* is a part of the electronic payment system, which acts as a interface between the user and the electronic payment system. It is assumed that the payment system leaves some flexibility for adapting the behavior of *PS*. This behavior will be detailed in the following sections. The authentication between a user and *PS* also depends on the electronic payment system used. The credentials, which are required for such purpose, have been assumed as an asymmetric key pair and its public key certificate. Following the notation used in this work, the user's credentials are (Pk_{py},Sk_{py}) and $cert_{py}$, and *PS*'s credentials are (Pk_{PS},Sk_{PS}) and $cert_{PS}$; and finally, *Punishment Authority PA*, which is supported by a competent law enforcement authority and com-

posed of a set of m sub-entities (or sub-authorities), is able to recover and reveal the identity of the owner of the V in case of fraud.

5.2.2 Requirements

The system requirements are related to fraud, privacy, authenticity and functionality, and they are described below in order to establish the foundations of the system.

Anti-fraud Requirements.

When a V is in the entrance of a *stretch*, it needs to obtain a **valid ticket** in order to properly enter it. *TP* issues *tickets* and requires V a **valid payment**. Otherwise, *TP* does not issue a *ticket*.

A **payment**, which is generated by *PS* and requested by *TP* in order to issue a *ticket*, is considered **valid** when (i) it cannot be modified once generated without detection (*integrity*); (ii) the *PS* can prove that this proof belongs to it (*authenticity*); (iii) *PS* cannot deny its authorship (*non-repudiation*); (iv) the amount of money required to be paid by V is fairly calculated by *TP*; and also, (v) it has not been reused (*uniqueness*).

A **payment** is **fair** when (i) the *prices* used to compute the amount of money required to pay, which are fixed by *SP*, are *valid*; (ii) the amount of money required to be paid is *correctly calculated* according to the emission category of V, the stretch and *valid prices*;

The **prices** of a stretch ($\alpha_{str}{}^{*}$), which are fixed by *SP*, are considered **valid** when (i) they cannot be modified once generated without detection (*integrity*); (ii) *SP* can prove that prices belong to it (*authenticity*); (iii) *SP* cannot deny its authorship (*non-repudiation*); (iv) they have not exceeded a fixed time after their generation (*temporality*); (v) and, they belongs to a certain stretch (*stretch belonging*)

A **ticket** is considered **valid** when (i) it cannot be modified once generated without detection (*integrity*); (ii) the issuer *TP* can prove that the ticket has been issued by it (ticket authenticity); (iii) the V can prove that the ticket belongs to it (ownership); (iv) they cannot deny its authorship (*non-repudiation*); (v) it has not exceeded a fixed time after its generation (*temporality*); (vi) it is associated to a certain *stretch* of the *LEZ* (*stretch belong-*

ing); and (vii) it is associated to a *valid payment*. *Tickets* contain information to prove that a specific *V* (owner) has the right to enter a certain *stretch*. The ticket is linked to a *V* and a validity time *ts*, which guarantee that the ticket can be used neither by another *V'*, nor after this period of time.

Fraud is committed by a *D* when she drives in a *stretch* of a *LEZ* (i) *without a ticket*; (ii) with an *invalid ticket*; (iii) with a *valid ticket but associated with another* V/stretch; or (iv) with a *valid ticket, which has been previously used to access the stretch (reused)*.

A false accusation takes place when a *SP* unjustly claims that a *V* (i) have *no ticket*; (ii) has an *invalid ticket*; or (iii) has a *valid ticket that belongs to another* V. On the contrary, a *V* cannot **falsely accuse** an honest *SP* of fraud.

Entity Authenticity Requirements.

The entities involved in the system exchange information. When the communication is established, each entity must prove their identity to the other part. In this way, each entity can be sure that the protocol is executed with the right one. If this is not the case, this action must be reported.

Privacy Requirements.

The fraud control executed by *SP* can endanger the privacy of the *D*s. In this case, the curiosity of *SP* could cause an excessive monitoring of the system or could even trace the itinerary of a specific *V*. With the aim of avoiding this excessive control by *SP* over the *V*s, the system must (i) assure the **anonymity** (the identity of *D* or *V* cannot be linked to any itinerary); (ii) avoid the **linkability** between itineraries (*SP* must not relate more than one itinerary to a same *V*); and (iii) provide **revocable anonymity** to *D* (if a *D* commits fraud, *SP* must know the identity).

Functional Requirements.

The communication between *V*s and *Chp*s/ *beacon*s needs to (i) be *started* when a *V* is close to a *Chp*s/*beacon*s, (ii) be carried out when a *V* is *moving*, and (iii) let *Chp* communicate with the *nearest V*. The communication has to start when a *V* is detected near to a *Chp*s/*beacon*s, for example, by broadcasting the information required to establish the communication. This

broadcast could be carried out by the use of Bluetooth Low Energy technology. Moreover, the communication in movement ([41]) can be possible by combining low and medium distance communication technologies, such as Wimax, ZigBee IEEE 802.15.4 or Bluetooth IEEE 802.15.1, using directional antennas or triangulation. Finally, the communication technology and the computation required by the protocol must be quick enough for a *Chp* and a *V*s to communicate between them without stopping. In addition, a fast mobile broadband communication is required for *V*s and *TP*/*PS* to communicate between them. Any interaction with the *D* should be easy and agile. The *Payment Service* should be anonymous and quick enough to allow the transaction when the *V* enters the *stretch*.

5.3 General Overview

The main purpose of a *LEZ* is to reduce the emissions of a urban zone with a tax payment. In this proposal, *LEZ*s are composed of a set of stretches. Figure 5.1 shows an example of a *LEZ* composed of stretches.

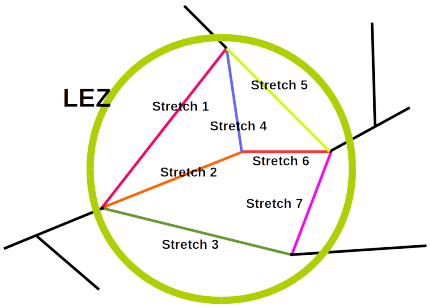


Figure 5.1: Example of a composition of a LEZ

Hence, the goal of this work is to allow the distribution of the traffic in the *LEZ* with the aim of reducing the travel time of each *V* as well as traffic jams. This is achieved with the dynamic assignment of prices for each stretch. As the traffic conditions are constantly changing, prices have to be adapted to such conditions. In such a way, every certain period of time, *SP* prices each stretch (see Section 5.6.1 Price generation), per emission category and depending on its traffic density. Therefore, *D*s are suggested to re-plan their route in order to cover less expensive routes (note that it depends on the distance to be covered, the consumption of fuel and the prices of the

stretches).

In order to achieve it, the system needs (i) to be initialized in Setup Protocol (Section 5.4); (ii) the certification of V and the generation of new credentials in Certification Protocol (Section 5.5); (iii) Vs to obtain a ticket through a payment in order to enter each stretch. The Payment Protocol (Section 5.6) includes the price of stretch and the way in which a V obtains a ticket. (iv) Vs to deliver a ticket in order to accesses the stretch in the *Entrance Protocol* (Section 5.7); (v) verifications in order to detect and sanction Ds that misbehave in Off-line Protocol (Section 5.8). Henceforth, these protocols are overviewed below:

Setup Protocol: Before starting the system, the entities *PA*, *VCA*, *SP*, *TP*, *Chp*s and *Beacon*s are initialized (Section 5.4)

Certification Protocol: *SE*s are initialized with a certification entity (see Section 3.2.2: Certificate Entity Installation). Afterwards, *SE* generates different credentials for V (see Section 3.2.3: Certificate Generation) every time it enters a *LEZ* (i.e. when the vehicle's engine is started up or just when the D is aware of her intentions of entering a *LEZ*) in order to correctly authenticate with other entities and to avoid linkability between their itineraries.

Payment Protocol: *Beacons*, placed at the beginning of the payment area of stretches, constantly send a warning message *information-of-stretch* (β_{str}). A vehicle V, when entering a payment area of a stretch *str* (see Section 5.6.2: Ticket Acquisition), receives a warning message from a *beacon* containing information of the current stretch. V then obtains the current prices of the stretch from *TP*, which has previously generated by *SP* for each stretch (see Section: 5.6.1 Price generation). According to the prices and its emission category, V makes a payment through *PS* and obtains a *proof-of-payment*. V then sends it to *TP*, which verifies the payment and issues a *ticket* (θ) associated with the V, the stretch and an expiration time.

Entrance Protocol: When a *Chp* placed at the beginning of the restricted area detects V (Section 5.7), which is getting close and which is in possession of a valid ticket, it communicates with V. They then mutually authenticate and V sends the ticket. When the authentication with V and the verification of the ticket fail, and only in this situation, *Chp* takes a picture of the V license plate as evidence of the infringement in order

to generate an incidence. This proof is sent to *PA* in order to verify the existence of fraud and so as to proceed with the corresponding sanction. On the other hand, when they are correct, *V* receives a *proof-of-entrance* as a receipt, and enters the restricted area of the stretch without being photographed.

While *V* remains in the *LEZ*, every time *V* enters a new stretch, the phase Ticket Acquisition (Section 5.6.2) of the Payment Protocol and the Entrance Protocol (Section 5.7) are repeated. Figure 5.2 shows the way in which a *V* obtains a ticket and enters the *stretch*.

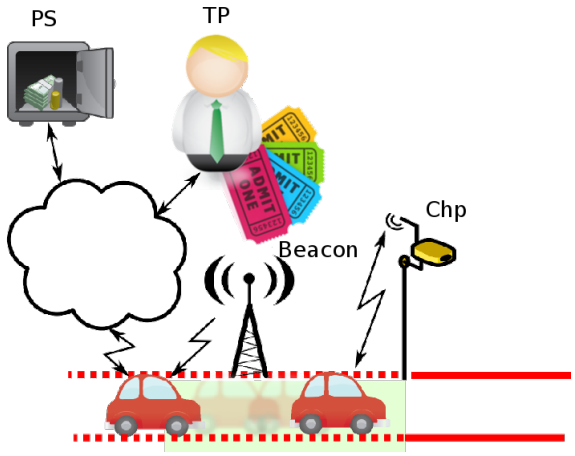


Figure 5.2: Entrance and Ticket Acquisition scheme

Off-line Protocol: The detection of reused tickets are performed every certain period of time (see Section 5.8.1: Doublespending Verification). Afterwards, *PA* verifies the incidences received (see Section 5.8.2: Sanction Process). Evidences to refute the accusation are then requested to *V* by *SP*. In case of fraud, *PA* reveals the identity of *V* (anonymity is revoked) and sanctions the owner.

The notation used and the proofs generated in this scheme are presented in tables 5.3 and 5.3, respectively.

Notation	Name	Description
(Pk_e, Sk_e)	Asymmetric key pair, public and private, belonging to entity e	
$Enc_e(m)$	Encryption of the message m using the asymmetric public key Pk_e	$Enc_e(m) = E_{Pk_e}(m) = m'$
$Dec_e(m')$	Decryption of the encrypted message m' using the asymmetric private key Sk_e	$E_{Sk_e}(m') = E_{Sk_e}(E_{Pk_e}(m)) = m$
$cert_e$	Public key certificate of the Pk_e of entity e supplied by CA_i	It includes Pk_e, $sign_{CA_i}(h(Pk_e\|...))$, etc.
$fing_e$	Fingerprint of the public key certificate Pk_e of entity e	It is computed by using a Hash function of the public key certificate
$cert_e.ext$	Certificate extension ext of the public key certificate $cert_e$	
$Sign_e(m)$ or $\bar{m}$	Digital signature of the message m by entity e	$Sign_e(m)= E_{Sk_e}(h(m)) = \bar{m}$
$Verif_e(m, \bar{m})$	Verification of the digital signature $\bar{m}$ by entity e	$Verif_e(m, \bar{m}) = E_{Pk_e}(\bar{m}) = E_{Pk_e}(E_{Sk_e}(h(m))) = h(m)?$
m^*	Set of infomation composed of the message and its signature	$m^* = (m, \bar{m})$
$h(m)$	Hash image of the message m	
N_x	Nonce X	It is a random number used in the authentication

Table 5.1: Notation

Proofs	Name	Content
α_{str}	*information-of-prices*	α_{str}=(str, $prices_{cat}$, p_{exp}, acc_d))
β_{str}	*information-of-stretch*	Information of the street stretch the V enters and information of TP connection which defines how to access TP
γ	*ticket-request*	γ = (REQ, N_A, str')
δ	*payment-request*	δ = (REQ, N_A, str', N_B, ts, $ticket_{id}$, $fing_{V_q}$, $amount$, $\overline{\alpha_{str}}$)
ϵ	*payment-order*	ϵ = (ACK, N_C, $fing_{PS}$, acc_s, $amount$, acc_d, REF)
ζ_1	*transfer-proof-1*	ζ_1 = ($amount$, acc_d, REF)
ζ_2	*transfer-proof-2*	ζ_2 = (acc_s, $amount$, acc_d)
η	*payment-proof*	η = (N_B, $\zeta_1{}^*$)
θ	*ticket*	θ = ($ticket_{id}$, str, ts, $fing_{V_q}$, η^*)
ι	*entrance-warning*	ι = (PET, N_D)
λ	*ticket-proof*	λ = (PET, N_D, N_E, $fing_{Chp}$, θ^*)
μ_i	*proof-of-entrance-incidence*	μ_i=(in_i, plt, ph, ts', str', λ^*, $cert_{V_q}$)
μ_v	*proof-of-verification-incidence*	μ_v = (in_v, (ν^*, $cert_{V_q}$), ($\nu^{*\prime}$, $cert_{V_q}$), ($\nu^{*\prime\prime}$, $cert_{V_q}$), etc.)
ν	*proof-of-entrance*	ν=(ts', N_E, $fing'_{V_q}$, λ^*)

Table 5.2: Proofs and data generated in the execution of the protocol

5.4 Setup Protocol

During the setup protocol, the following entities are initialized:

1. *PA*: This entity is constituted by a set of m members. An asymmetric key pair (Pk_{PA}, Sk_{PA}) is securely generated with a threshold scheme such as in [36]. The Sk_{PA} is divided into m pieces or shares and is securely distributed among the m entities. From competent authorities (i.e. Police) they obtain its public key certificate $cert_{PA}$, and a certificate repository of the authorities

2. *VCA* generates and associates, as it is detailed in Section 3.2.1, a certification authority VCA_{C_i} for each group of users, that is, for each element of the subset $K = \{C_1, C_2, ..., C_{num_K}\}$, consisting of an asymmetric key pair ($Pk_{VCA_{C_i}}$, $Sk_{VCA_{C_i}}$), and CA certificate $cert_{VCA_{C_i}}$

3. *SP*, *TP* and each *Chp* apply the following steps:

   i. Securely generate an asymmetric key pair (Pk_{SP}, Sk_{SP}), (Pk_{TP}, Sk_{TP}) and (Pk_{Chp}, Sk_{Chp}), respectively

   ii. Securely obtain the public key certificate ($cert_{SP}$, $cert_{TP}$ and $cert_{Chp}$) from the competent authorities (i.e. city council). The $cert_{SP}$ validity period could correspond to the concession lifetime of the service, without exceeding it. Furthermore, $cert_{Chp}$ contains an extension $cert_{Chp}.loc$ with its location coordinates and a stretch identifier $cert_{Chp}.str$, which links the *Chp* with a certain stretch

   iii. Securely obtain a certificate repository of the authorities

4. *Beacons* are initialized by *SP* with a warning advise *information-of-stretch* $\beta_{str}{}^{*}$=(β_{str},$\overline{\beta_{str}}$), where $\overline{\beta_{str}}$ is the signature of $\beta_{str}(Sign_{SP}(\beta_{str}))$ and where β_{str} contains:

   - Information of the stretch the V enters (such as *str*, GPS location coordinates, etc.)
   - Information of *TP* connection which defines how to access *TP* (through 3G, ZigBee or other technologies)

5.5 Certification Protocol

This protocol consists of two phases: In the first phase, a certification entity is installed in the *SE* of a *V* by *VCA*. In the second phase, the *SE*, which has been previously initialized, generates new credentials every time it enters a *LEZ*.

5.5.1 Certificate Entity Installation

In this phase, *Vehicle Certification Authority VCA* registers a user in a group of users (an element of the subset *K*) and securely installs the corresponding Certification Authority VCA_{C_i} associated to C_i in the *SE* of each *V*, consisting of $Pk_{VCA_{C_i}}$, $Sk_{VCA_{C_i}}$ and $cert_{VCA_{C_i}}$. This phase is performed from time to time, for example, before purchasing a vehicle and/or passing the regular technical vehicle tests. Check Section 3.2.2 for further details.

5.5.2 Certificate Generation

The *SE* of the *V*, which has a certification entity VCA_{C_i} installed, generates new credentials. The resulting credentials consist of an asymmetric key pair (Pk_{V_q}, Sk_{V_q}) and a public key certificate $cert_{V_q}$ containing an extension $cert_{V_q}.idS$, which is the encryption of the vehicle identifier V_{id} $Enc_{Pk_{PA}}(V_{id})$. This phase is performed every time a *V* is on its way to enter a *LEZ* in order to avoid linkability between its trips. For example, when the vehicle's engine is started up or just when the *D* decides enter a *LEZ*. Check Section 3.2.1 for more information about this phase.

The repetitive execution of this phase can produce a a large amount of data since many asymmetric key pairs and their public key certificates are generated. Considering the limited storage capabilities of *SE*s, it could be interesting to securely store the information concerning to previous *LEZ* entrances in the *OBU* by means of key management custody techniques ([38, 39]).

5.6 Payment Protocol

In this section, the functional part of the system where users pay for accessing a traffic restricted zone is described. The payment protocol is described

in the following two subsections: *price generation* and *ticket acquisition*.

5.6.1 Price Generation

Every fixed period of time, *SP* establishes the prices of each stretch *str* by performing the next operations:

1. Set the prices per emission category $prices_{cat}$ (i.e. European Emission Standards), searching a balance between supply and demand. The supply is the number of vehicles that a stretch can contain. It depends on the stretch capacity (the number of lanes and their length) and the current occupancy, which can be counted in various ways (such as with an induction loop or by accounting the entrance of the next stretch). The demand is variable and depends on the number of vehicles that want to enter the stretch. The calculation to find a right price, where demand is unknown in advance, is a known issue in other fields such as mobile networks. [47] is an example of a calculation method, which predicts the demand and could be easily adapted to the present context.

2. Compose *information-of-prices* α_{str}=(str, $prices_{cat}$, p_{exp}, acc_d), where p_{exp} is the expiration time and acc_d identifies the *SP* destination account of the electronic payment system.

3. Sign α_{str}: $Sign_{SP}(\alpha_{str}) = \overline{\alpha_{str}}$, send ${\alpha_{str}}^*$ to *TP*

5.6.2 Ticket Acquisition

In this phase, the acquisition of a ticket by a *V*, before entering a *stretch*, is described. Therefore, when *Beacons*, which are placed at the beginning of the payment area of the stretches, detects a *V*, the following steps are applied:

1. The *Beacon* sends *V* an *information-of-stretch* ${\beta_{str'}}^*$

2. The SE of the *V*, with the help of the *OBU*, has to:

   i. Verify the signature $\overline{\beta_{str'}}$: $Verif_{SP}(\beta_{str'}, \overline{\beta_{str'}})$

   ii. Verify GPS location coordinates included in $\beta_{str'}$.

   iii. Extract $str'$from $\beta_{str'}$

iv. Generate a nonce N_A and a ticket request REQ

v. Compose a *ticket-request* $\gamma = (REQ, N_A, str')$

vi. Sign γ: $Sign_{V_q}(\gamma) = \overline{\gamma}$, and send γ^* and its certificate $cert_{V_q}$ to TP

3. TP has to:

i. Verify the certificate $cert_{V_q}$ and the signature $\overline{\gamma}$: $Verif_{SP}(\gamma, \overline{\gamma})$

ii. Generate a nonce N_B and a time ts, increment a counter $ticket_{id}$, and compute the $fing_{V_q}$ of $cert_{V_q}$

iii. Recover ${\alpha_{str}}^*$ of the requested stretch and obtain $amount$ from $prices_{cat}$ according to its pollutant emissions $cert_{V_q}.em$

iv. Compose a *payment-request* $\delta = (REQ, N_A, str', N_B, ts, ticket_{id}, fing_{V_q}, amount, \overline{\alpha_{str}})$

v. Sign δ: $Sign_{TP}(\delta) = \overline{\delta}$, and send N_B, $ticket_{id}$, ts, $amount$, $\overline{\delta}$ and ${\alpha_{str}}^*$ to V

4. The SE of the V, with the help of the OBU, has to:

i. Verify the signature $\overline{\alpha_{str}}$: $Verif_{SP}(str', prices_{cat}, p_{exp}, acc_d, \overline{\alpha_{str}})$, and verify the freshness of ${\alpha_{str}}^*$:$|p_{exp}$-*current time*$| < \tau$, where τ is a fixed time interval

ii. Obtain the $amount'$ of money required to pay according to the pollutant emissions $cert_{V_q}.em$ of V and the current $prices_{cat}$, included in α_{str}

iii. Verify the signature $\overline{\delta}$: $Verif_{PS}(REQ, N_A, str', N_B, ts, ticket_{id}, fing_{V_q}, amount', \overline{\alpha_{str}}, \overline{\delta})$, and verify the freshness of ts: $|ts$-*current time*$| < \tau'$, where τ' is a fixed time interval

iv. Access PS and establish a secure communication channel (i.e. by means of TLS/SSL protocol)

5. PS generates a nonce N_C, and sends it to V

6. The SE of the V, with the help of the OBU, has to:

i. Compute the $fing_{PS}$ of $cert_{PS}$

ii. Compute a reference $REF = hash(\overline{\delta})$

iii. Recover the destination account acc_d and $prices_{cat}$ from α_{str}

iv. Compose a *payment-order* $\epsilon = (N_C, fing_{PS}, acc_s, amount, acc_d, REF)$, where acc_s is the source account of the user

v. Sign[1] ϵ: $Sign_{py}(\epsilon) = \overline{\epsilon}$

vi. Send acc_s, $amount$, acc_d, REF, $\overline{\epsilon}$ and the certificate $cert_{py}$ to PS

7. *PS* has to:

i. Verify the certificate $cert_{py}$ and the signature $\overline{\epsilon}$: $Verif_{py}(\epsilon, \overline{\epsilon})$

ii. Extract acc_s, $amount$ and acc_d from ϵ and make a digital transfer of $amount$ from the account acc_s to the account acc_d

iii. Compose a *transfer-proof-1* $\zeta_1 = (amount, acc_d, REF)$

iv. Compose a *transfer-proof-2* $\zeta_2 = (acc_s, amount, acc_d)$

v. Sign ζ_1: $Sign_{PS}(\zeta_1) = \overline{\zeta_1}$, and sign ζ_2: $Sign_{PS}(\zeta_2) = \overline{\zeta_2}$

vi. Send $\overline{\zeta_1}$ and $\overline{\zeta 2}$ to the *payer*

vii. Send ${\zeta_2}^*$ to *SP* every certain period of time, in batch, jointly with other ${\zeta_2}^*$s from other users and transactions

8. The *SE* of the *V*, with the help of the *OBU*, has to:

i. Verify the signature $\overline{\zeta_2}$: $Verif_{PS}(acc_s, amount, acc_d, \overline{\zeta_2})$

ii. Verify the signature $\overline{\zeta_1}$: $Verif_{PS}(amount, acc_d, REF, \overline{\zeta_1})$

iii. Compose a *payment-proof* $\eta = (N_B, {\zeta_1}^*)$

iv. Sign η: $Sign_{V_q}(\eta) = \overline{\eta}$, and send $\overline{\eta}$ and $\overline{\zeta_1}$ to *TP*

9. *TP* has to:

i. Verify the signature $\overline{\eta}$: $Verif_{V_q}(N_B, {\zeta_1}^*, \overline{\eta})$

ii. Compute the reference $REF' = hash(\overline{\delta})$

iii. Verify the signature $\overline{\zeta_1}$: $Verif_{PS}(amount, acc_d, REF', \overline{\zeta_1})$

iv. Compose a *ticket* $\theta = (ticket_{id}, str, ts, fing_{V_q}, \eta^*) = (ticket_{id}, str, ts, fing_{V_q}, N_B, amount, acc_d, REF, \overline{\zeta_1}, \overline{\eta})$

v. Sign θ: $Sign_{TP}(\theta) = \overline{\theta}$, and send $\overline{\theta}$ to V

[1]Note that the payment of a *stretch* is made by the user or *D* with its payment credentials, which depends on the electronic payment

10. The *SE* of the V, with the help of the *OBU*, verifies the signature $\overline{\theta}$: $Verif_{TP}(ticket_{id}, str, ts, fing_{V_q}, \eta^*, \overline{\theta})$

5.7 Entrance Protocol

The entrance process takes place when a V is approaching a restricted area after passing the payment zone. A *Chp*, which is placed at the beginning of the restricted area of a *stretch*, then requires the V to deliver a *ticket* in order to get into it. The following steps are applied when a *Chp* detects a V:

1. *Chp* has to:
   i. Generate a nonce N_D and a ticket petition PET
   ii. Compose an *entrance-warning* $\iota = (PET, N_D)$
   iii. Sign ι: $Sign_{Chp}(\iota) = \bar{\iota}$, and send ι^* and its certificate $cert_{Chp}$ to V
2. *SE* of the V, with the help of the *OBU*, has to:
   i. Verify the certificate $cert_{Chp}$, the GPS location coordinates $cert_{Chp}.loc$ and the stretch identifier $cert_{Chp}.str = str$, which is included in θ
   ii. Verify the signature $\bar{\iota}$: $Verif_{Chp}(\iota, \bar{\iota})$
   iii. Generate a nonce N_E and compute the $fing_{Chp}$ of $cert_{Chp}$
   iv. Compose a *ticket-proof* $\lambda = (PET, N_D, N_E, fing_{Chp}, \theta^*)$
   v. Sign λ: $Sign_{V_q}(\lambda) = \overline{\lambda}$, and send N_E, θ^*, $\overline{\lambda}$ and its $cert_{V_q}$ to *Chp*
3. *Chp* has to:
   i. Generate a time ts'
   ii. Verify the certificate $cert_{V_q}$ and the signature $\overline{\lambda}$: $Verif_{V_q}(N_D, fing_{Chp}, N_E, \theta^*, \overline{\lambda})$
   iii. Recover $cert_{Chp}.str$ and compute the fingerprint $fing'_{V_q}$ of $cert_{V_q}$
   iv. Verify the signature $\overline{\theta}$: $Verif_{TP}(ticket_{id}, cert_{Chp}.str, ts, fing'_{V_q}, \eta^*, \overline{\theta})$
   v. Verify $cert_{Chp}.str = str$, which is included in θ
   vi. Verify the freshness of θ^*:$|ts - ts'| < \tau''$, where τ'' is a time fixed interval according to the traffic volume of the stretch.

vii. If one of the verifications fails or *ticket-proof* has not sent, *Chp* performs the following operations:

vii.i Generate an incidence number of entrance in_i

vii.ii Take a photo ph of V and extract the plate number plt

vii.iii Compose a *proof-of-entrance-incidence* μ_i=(in_i, plt, ph, ts', str', λ^*, $cert_{V_q}$)

vii.iv Sign μ_i: $Sign_{Chp}(\mu_i) = \overline{\mu_i}$, and send ${\mu_i}^*$ and $cert_{Chp}$ to *SP* through a secure channel

viii. If the verifications performed in 3ii-3vi are correct, *Chp* has to:

viii.i Compose *proof-of-entrance* ν=(ts', N_E, $fing'_{V_q}$, λ^*)

viii.ii Sign ν: $Sign_{Chp}(\nu) = \overline{\nu}$, and send ts' and $\overline{\nu}$ to the V

4. If the verifications performed in 3ii-3vi are correct, *SE* of the V, with the help of the *OBU*, has to:

   i. Verify the signature $\overline{\nu}$: $Verif_{Chp}(ts', N_E, fing'_{V_q}, \lambda^*, \overline{\nu})$

5.8 Off-line Protocol

This protocol is composed of two phases. Both are performed in batch after a certain period of time. In the first phase, *SP* detects whether fraud is committed by reused tickets. In the second one, *PA* verifies all the incidences generated by the systems and sanctions the Vs that misbehave.

5.8.1 Doublespending Verification

Each *Chp* periodically sends *SP* the pair *proof-of-entrance*s ν^* and the associated certificate of vehicle $cert_{V_q}$, and also the *proof-of-entrance-incidence*s ${\mu_i}^*$, which are generated in Section 5.7. Then *SP* forwards the incidences ${\mu_i}^*$ to *PA* through a secure channel. Moreover, *SP* performs the next operations, in batch, every certain period of time:

1. Define a set of pairs of *proof-of-entrance* $I = \{({\nu_1}^*, cert_{V_1}), ({\nu_2}^*, cert_{V_2}), ..., ({\nu_{n_\nu}}^*, cert_{V_{n_\nu}})\}$, where n_ν is the number of *proof-of-entrance* sent to *TP*

2. Verify that there is a unique ν in the set I with the same θ^*

3. If more than a v with the same θ^* are found ($v^{*\prime}$, $v^{*\prime\prime}$, ...), *SP* then needs to:

   i. Generate an incidence number of verification in_v

   ii. Compose *proof-of-verification-incidence* μ_v including the concerned *proofs-of-entrance* v^*: $\mu_v = (in_v, (v^*, cert_{V_q}), (v^{*\prime}, cert_{V_q}), (v^{*\prime\prime}, cert_{V_q}), ...)$, which proves the doublespending

   iii. Sign μ_v: $Sign_{SP}(\mu_v)=\overline{\mu_v}$, and send ${\mu_v}^*$ to *PA* through a secure channel

5.8.2 Sanction Process

This section is the last phase of the system, where incidences generated by *SP* are evaluated in order to sanction dishonest users. The *Punishment Authority PA* performs the following sanction process:

For each received μ^*, the following operations are performed:

1. In the case of ${\mu_i}^*$:

   i. The m sub-entities of *PA* perform the following operations:

      i.i. Verify $cert_{Chp}$ and $cert_{V_q}$

      i.ii. Compute the fingerprint $fing''_{Chp}$ of $cert_{Chp}$ and the fingerprint $fing''_{V_q}$ of $cert_{V_q}$

      i.iii. Recover $cert_{Chp}.str$

      i.iv. Extract the plate number plt' of the photo ph

      i.v. Verify the signature $\overline{\mu_i}$: $(in_i, plt', ph, ts', cert_{Chp}.str, \lambda^*, cert_{V_q}, \overline{\mu_i})$

      i.vi. Verify the signature $\overline{\lambda}$: $Verif_{V_q}(N_D, fing''_{Chp}, N_E, \theta^*, \overline{\lambda})$

      i.vii. Verify the signature $\overline{\theta}$: $Verif_{TP}(ticket_{id}, cert_{Chp}.str, ts, fing''_{V_q}, \eta^*, \overline{\theta})$

      i.viii. Verify the freshness of θ^*:$|ts - ts'| < \tau''$

      i.ix. If one of the previous verifications fails, that is, the incidence is confirmed by a least t from m members of *PA*, the $fing_{V_q}$ of $cert_{V_q}$ is then made public. An example could be using a *Bulletin Board*, which is widely used in many countries such as the "Tablón Edictal de Sanciones de Tráfico" of the Spanish

Ministry of Internal Affairs[2]. In a private zone of the *Bulletin Board*, the proofs presented by the *SP* are uploaded. Note that, at this point, some of the considered verifications might not be performed since the required proofs might not be sent or generated by V. This case is considered and dealt with as an incidence.

ii. The owner of the V performs the following operations:

ii.i. Look for fingerprints of her $Cert_{V_q}$ in the *Bulletin Board* every certain period of time

ii.ii. If one of her fingerprints is found on the Bulletin Board

ii.ii.i. Obtain more information of the sanction process by accessing the private zone of the *Bulletin Board*. The access to the private bulletin board is controlled by the authentication with the credentials Sk_{V_q} and $cert_{V_q}$.

ii.ii.ii. Send evidences to *PA* in order to refute her accusation. These evidences, which could be proofs generated in the execution of the protocol, will depend on the previous unsuccessful verifications.

iii. The m sub-entities of *PA* have to:

iii.i. Verify the evidences presented by the owner of the V to refute the accusation

iii.ii. If the presented contra evidences are not considered valid by at least t from m members of *PA*, they have to:

iii.ii.i. Extract the number plate plt from the photograph ph

iii.ii.ii. Identify the owner of the V by using plt

iii.ii.iii. Fine the owner of V according to the type of infraction

2. In the case of $\mu_v{}^*$:

i. The m sub-entities of *PA* perform the following operations:

[2]https://sede.dgt.gob.es/es/tramites-y-multas/alguna-multa/consulta-tablon-edictal-testra/

i.i. Verify the signature $\overline{\mu_v}$: (in_v, (v^*, $cert_{V_q}$), ($v^{*\prime}$, $cert_{V_q}$), ($v^{*\prime\prime}$, $cert_{V_q}$), ..., $\overline{\mu_v}$)

i.ii. For each pair (v^*, $cert_{V_q}$):

ii.ii.i Verify $cert_{V_q}$

ii.ii.ii Compute the fingerprint $fing''_{V_q}$ of $cert_{V_q}$

ii.ii.iii Verify the signature $\overline{v}$: $Verif_{Chp}(ts', N_E, fing''_{V_q}, \lambda^*, \overline{v})$

ii.ii.iv Verify the signature $\overline{\lambda}$: $Verif_{V_q}(N_D, fing_{Chp}, N_E, \theta^*, \overline{\lambda})$

ii.ii.v Verify the signature $\overline{\theta}$: $Verif_{TP}(ticket_{id}, str, ts, fing''_{V_q}, \eta^*, \overline{\theta})$

i.iii. Verify that each v has the same θ^*

i.iv. If all the verifications are correct, that is, the incidence is confirmed by a least t from m members of *PA*, the $fing_{V_q}$ of $cert_{V_q}$ is made public, for example in a *Bulletin Board*. The proofs presented by the *SP* are uploaded in a private zone of the *Bulletin Board*.

ii. The owner of V performs the following operations:

ii.i. Look for fingerprints of her $Cert_{V_q}$ in the *Bulletin Board* every certain period of time

ii.ii. Obtain more information of the sanction process, through the authentication with her credentials (Sk_{V_q} and $cert_{V_q}$), if one of her fingerprints is found on the Bulletin Board, and get access to the private zone of the *Bulletin Board*

ii.iii. Send evidences to *PA* in order to refute her accusation. These evidences, which could be proofs generated in the execution of the protocol, will depend on the previous unsuccessful verifications.

iii. The m sub-entities of *PA* have to:

iii.i. Verify the evidences presented by the owner of the V to refute the accusation

iii.ii. If the presented contra evidences are not considered valid by at least t from m members of *PA*, they have to:

iii.ii.i. Partially open the extension $certV_q.idS$ of the certificate $certV_q$, which is included in λ, with their own shares Sk_{PA_m}: $Dec_{PA_m}(certV_q.idS) = V_{id_m}$

iii.ii.ii. Obtain the identifier of the owner of the V (V_{id}) by aggregating all V_{id_m}

iii.ii.iii. Fine the owner of V according to the type of infraction

After the sanction process, if the proofs and the data, which were generated in the previous protocols (Section 5.7 and 5.6) such as θ^* and λ^*, have been subject to no sanction, they can be deleted. It can be particularly interesting for *V*s since the *OBU*'s storage is not unlimited.

5.9 Security and Privacy Analysis

In this section, the adversary model and the analysis of security and privacy of the proposed system are presented. In particular, the anti-fraud, authenticity and privacy requirements are evaluated. that were introduced in Section 5.2.2.

5.9.1 Adversary Model

Taking into account the entities involved in the proposed architecture and the existent interactions between them, the following adversaries have been considered:

- *Dishonest driver.* There could be drivers interested in avoiding fee payment. Dishonest drivers can take action against the system by modifying the *OBU*, which is installed in the *V* and is accessible by the *D*. The modification of the *OBU* is the gateway to all the attacks on the system promoted by *D*s. Note that, a *D* capable of tampering with an *OBU*, may illegally obtain significant benefits from the system. This act could be detected thanks to the physical seal on the *OBU*, which is checked by the fully trusted entity *VCA* (it is maintained by the competent traffic authority). Despite this, this seal does not prevent attackers from tampering with *OBU*s. If a *D* modifies the *OBU*, she will have a period of time until the next certification, which could be spent to take benefit

in some way. This is not the case of the *SE*, which is also managed by the *VCA* and is a tamper proof device. The sealing period of time should be short and controllable in the certification process, where the installation of the new CA credentials in the *SE* is made physically and in person by a *VCA*'s operator. Note that these credentials are an essential requirement to circulate in the *LEZ* without being fined. A *V* with no credentials or obsolete credentials will be identified, losing its privacy, and being finally fined. A part from that, other measures to prevent possible attacks (such as fraud) will be detailed in this section.

- *Dishonest Service Provider.* Monitoring all vehicle movements and identifying both vehicles and drivers can be useful for the *SP*, because the system can know when fraud is committed as well as its authorship. In this way, the system safeguards its interests, punishing *dishonest drivers*, although the consequence of this can affect the privacy rights of the driver. Moreover, *SP* could groundlessly accuse some *V* of a fraud with the desire of collecting more money, or just to know the real identity of *V*. The entities *Beacons*, *Checkpoints* and *Ticket Provider* could participate following the *SP* purposes since they are managed by it. For example, if the *SP* wants to monitor a *V*, the set of *Chps* could act to help it.

- *External Attacker.* This adversary may be an individual with no relation with the protocol. This attacker could perform passives attacks such as a communication eavesdropping on foot of street, or active attacks such as the modification of am *OBU* of any *V*, like in the case of a dishonest driver.

Despite the fact that fraud fighting and privacy protection can sometimes be contradictory objectives, the goal of the scheme presented in this work is to address both. It is also important to note that the system is not able to protect drivers' privacy against internal or external attackers, who follow *V*s along their trips. This system can also suffer from physical world problems, which can be solved with measures belonging to a more physical field. In the same way, the system proposed cannot fight against the installation of parallel cameras, which register *V*s passing through *Chp*s, or even against *Chps* taking photos of honest *V*s irregularly. In this case, *Chp*s would then

be acting against the protocol. As a result, these sneaky behaviors are not taken into account in the following security analysis.

Regarding other entities of the proposed system, the *PA* is considered a trusted entity because this entity and the set of m sub-entities that compose it are supported by a competent law enforcement authority. As if that was not enough, thanks to the use of a threshold scheme, whatever they do has to be hold by the majority of m, in this case, larger than t. For example, if a set of these m sub-entities was dishonest, the number of dishonest sub-entities should be larger than t to be able to take dishonest actions. For all these reasons, it is assumed that *PA* is considered a fully trusted entity. Furthermore, *PS* is also considered a trusted entity because the electronic payment system that manages it is, in essence, a trusted entity too.

5.9.2 Security Analysis

The security properties to be guaranteed by the system, which are stated in Section 5.2: System Model, are fraud, false-accusation and privacy. The security analysis of these properties is organized in several claims that support the fulfillment of lemmas and theorems.

Anti-Fraud Requirements

Claim 1. *The creation of a fraudulent* message or proof *is computationally unfeasible nowadays.*

Proof. A *message* is composed of a set of concatenated data. When the *issuer* of a *message* signs it using its asymmetric private key $Sign_{Issuer}$ (*message*) = $\overline{message}$, the generation of *ticket*s is computationally unfeasible nowadays without the knowledge of the *issuer*'s secret key used in the signature. It is assumed that the issuer securely stores its secret key. □

Claim 2. *An* issuer *of a* message or proof *cannot deny its emission.*

Proof. When a *message* is generated and signed by its *issuer* and, considering the signature scheme secure against current computational attacks, this operation can only be performed by the *issuer*. Thus, the issuer's identity is linked to the proofs and, for the properties of the electronic signature scheme, it cannot deny its authorship. □

Claim 3. *The content of a* message or proof *cannot be modified without detection.*

Proof. Let's suppose that the signature scheme is secure and that the hash summary function is collision-resistant. If the content of a signed *message* was modified, the verification of the signature would be incorrect because $Sign_{Issuer}(message) = E_{Sk_{Issuer}}(hash(message)) = \overline{message}$. In order to pass the verification, the signature would be regenerated from the new *message*. This operation is computationally unfeasible without the knowledge of the *issuer*'s secret key. It is assumed that the issuer securely stores its secret key. □

Claim 4. *A*message or proof *can be detected when it has expired.*

Proof. When a *message* containing an expiration time *ts* is generated and signed by its *issuer*, and since the *message* cannot be modified, as it is stated in Claim 3, the expiration of a message can be detected using *ts*. □

Claim 5. *A* message or proof *belongs to a stretch/vehicle.*

Proof. When a *message* containing an identifier of stretch/vehicle is generated and signed by its *issuer*, and since the *message* cannot be modified, as it is stated in Claim 3, the relation of stretch/vehicle to a message can be detected. □

Lemma 5.9.2.1 *The proposed system must preserve authenticity, non-repudiation, integrity, temporality, and stretch belonging for* prices *in order that the* prices *are* valid.

Proof. When the *issuer* of a message is the *SP*, the *message* is an *information-of-prices* $\alpha_{str}{}^{*}$ (where the prices are fixed), and the *str* is an identifier used to link a stretch with prices, *Claims 1, 2, 3, 4, and 5* imply that the properties of authenticity, non-repudiation, integrity, temporality, and stretch belonging hold for the *information-of-prices* $\alpha_{str}{}^{*}$. In the protocol, these properties are detected in step 4i of Section 5.6. Hence, the protocol satisfies the needed security requirements, and the *prices* are thus *valid*. □

Claim 6. *A* payment *is* fair.

Proof. The amount of money required to be paid by a V, which is contained in ${\zeta_1}^*$, is fair when (i) *TP* have calculated it taking into account *valid prices* and that prices belong to the stretch that V wants and to the *emission category* of V; and (ii) when V has verified it.

The amount of money is computed in step 3iii of Section 5.6 and it is sent by *TP*, jointly with *information-of-prices* ${\alpha_{str}}^*$, with a signed *payment-request* δ^*. Considering that (i) the validity of prices is proved in Lemma 5.9.2.1; (ii) the *payment-request* cannot be modified as it is stated in Claim 3; (iii) the issuer *TP* of the *payment-request* cannot deny its emission as it is stated in Claim 2; and (iv) the calculus can be verified by the V, before it pays, in step 4ii and 4iii of Section 5.6, the amount of money required to pay is therefore fair. □

Claim 7. Payment*s cannot be reused without detection.*

Proof. Considering that a ${\zeta_1}^*$ contains a $REF = hast(\underline{(}\delta))$ (see steps 6ii and 7iii of Section 5.6), and that δ is generated with random *nonces* from a V and *TP* among others, the provability to find a collision is negligible. Therefore, when a *payment-proof-1* ${\zeta_1}^*$ containing REF is generated and signed by *PS*, and since the *message* cannot be modified, as it is stated in Claim 3, the payment is unique. In the protocol, the detection is performed by *TP* in steps 9ii and 9iii of Section 5.6. □

Lemma 5.9.2.2 *The system complies with the authenticity, non-repudiation, integrity, fairly and uniqueness for* payments *in order that* payments *are* valid.

Proof. When the *issuer* of a message is the *PS* and the *message* is a *payment-proof-1* ${\zeta_1}^*$, *Claims 1, 2, 3, 6, and 7* imply that the properties of authenticity, non-repudiation, integrity, fairness and uniqueness hold for the *payment-proof-1* ${\zeta_1}^*$. In the protocol, authenticity, non-repudiation and integrity are detected in step 9iii of Section 5.6.

Hence, the protocol satisfies the needed security requirements and the *payment* is thus *valid*. □

Lemma 5.9.2.3 *The system complies with authenticity, non-repudiation, integrity, temporality, stretch belonging, ownership, and* valid payment *for* tickets *in order that* tickets *are* valid.

Proof. When the *issuer* of a message is the *TP*, *ticket* θ^* is considered the *message*, the *str* is an identifier used to link a stretch with prices, and the $fing_{V_q}$ is an identifier of a V used to link a V with a *ticket*, *Claims 1, 2, 3, 4, 5* and Lemma 5.9.2.2 imply that the properties of authenticity, non-repudiation, integrity, temporality, stretch belonging, ownership, and *valid payment* hold for the *ticket* θ^*. In the protocol, authenticity, non-repudiation and integrity are detected in steps 3ii-3vi of Section 5.7 by *Chp*.

Hence, the protocol satisfies the needed security requirements and the *ticket* is thus *valid*. □

Claim 8. *The toll system detects fraud when it is committed by a user.*

Proof. Fraud is detected when a user enters a stretch without a ticket, with an invalid ticket, with a valid ticket associated with another vehicle or stretch, or with a *valid ticket, which has previously been used to access the stretch (reused)*. In the first case, a driver attempting to enter a stretch with no ticket will be detected in step 3vi of Section 5.7. The detection of a user with no valid ticket is stated in Lemma 5.9.2.3. In the third case, a ticket associated with another stretch is detected in step 3v of Section 5.7, and associated with another user in step 3v of Section 5.7. Finally, in the last case, a *valid ticket, which has been previously used to access the stretch (reused)*, is detected in Section 5.8.1. □

Claim 9. *The system can identify dishonest users thanks to the anonymity revocation offered by the system.*

Proof. A dishonest user can be identified in two different ways according to how fraud is detected (stated in Claim 8).

When fraud is detected due to a *reused* ticket in Section 5.8.1, *SP* sends $\mu_v{}^*$ to *PA* to issue a traffic fine for the user. *PA* verifies the incidence and asks V for evidences. Note that these evidences are requested to V without being

identified thanks to the use of a *Bulletin Board*. If the evidences do not refute the accusation, then *PA* identifies the user by opening the field $certV_q.idS$ of the certificate with its private key. Obtaining V_{id} allows the identification and the punishment of the dishonest user.

In other cases of fraud, the user is identified when she is photographed by the *Chp* at the entrance of a stretch (it is described in step 3vi of Section 5.7. The user then loses her anonymity as the vehicle number plate is captured. This information is sent to *PA* to act as it is specified in the *Sanction* process. In the execution of this protocol, *PA* has the ability to identify the user through the number plate. □

Theorem 5.9.2.4 *The toll collection scheme keeps fraud under control and it can identify dishonest users. These users receive the appropriate traffic fine. In contrast, honest users are kept anonymous.*

Proof. As it is stated in Claim 8 and 9, the system can detect *fraud* and identify the fraudulent Vs. □

Claim 10. *The protocol execution generates evidences for an honest user (they are saved in the* OBU*) to prove the allegations of fraud.*

Proof. A user, after entering a stretch, has obtained and generated several messages ($cert_{V_q}$, ${\alpha_{str}}^*$, ${\beta_{str}}^*$, γ^*, δ^*, ϵ^*, ζ^*, η^*, θ^*, ι^*, λ^*, and ν^*) signed by different entities involved in the protocol (SP, TP, Chp and herself), which prove she has entered the stretch without committing fraud. *Claims 1, 2, and 3* imply that the properties of authenticity, non-repudiation and integrity hold for all these *messages*.

When a user is unjustly accused of fraud by *SP*, the user is notified by *PA* during the *Sanction* process. She will then be able to retrieve some of these records and provide them to *PA* in order to prove its own honesty (see steps of Section 5.8.2-1ii and 5.8.2-2ii according to incidence type). □

Claim 11. *The protocol execution generates evidences for an honest SP to prove the allegations of fraud.*

Proof. As in the case described in Claim 10, the system generates mes-

sages/proofs, which can be used by the *SP* to refute an accusation when it is unjustly accused of fraud by a *V*. All these proofs are generated and signed by entities of the system, which prove the correct functioning of the system. *Claims 1, 2, and 3* imply that the properties of authenticity, non-repudiation and integrity hold for all these *messages*. Therefore, these message are accepted as proofs. □

Theorem 5.9.2.5 *The system protects users and SP against false accusations.*

Proof. As it is stated in Claim 10 and Claim 11, the protocol allows honest users and SP to get evidences of their correct performance. The evidences are used to rebut any traffic fine due to some kind of malfunction of the system.□

Entity Authenticity Requirements

In order to prevent impersonation, all the entities involved in the system, which are not trusted, use authentication based on the access to a predistributed authentic material (public keys and CA certificates).

In each communication, each entity convinces its communication partner that, in the current communication, its identity is as declared. This is achieved by signing a message with its credentials (secret key). This entity then proves its identity by its signature on the message. If an adversary intercepts the message signed by the entity, it could use it later to authenticate herself as the entity. In order to prevent these attacks, which are called replay attacks, random numbers or nonces are used as a challenge-response scheme. The communication partner sends a nonce (challenge) to the prover entity and the prover then returns a signed message containing this nonce (response). In this way, the message signed varies each time (message uniqueness) and an attacker can reuse an old message.

In the communications where both entities are not trusted, the protocol follows a 3-way mutual authentication scheme, similar to the X.509 three-way authentication protocol.

In addition, the use of signatures and public key certificates prevent from Man in the Middle attacks.

In the particular case of the authentication of a V to another entity, the generation of the asymmetric key pair and the public key certificate by means of the CA installed in the *SE*, the signature of the messages sent by it, and the storage of these credentials are performed in the *SE* of the V. Therefore, the authentication of a V depends on how securely the initialization of the *SE* with a *CA* has been performed (as it is shown in Section 5.9.1) and how much secure a *SE* is.

Privacy Requirements

Claim 12. *The system does not allow to link an itinerary with a vehicle.*

Proof. Considering that an itinerary is composed of messages or proofs, which place a V in a set of ordered *stretches* such as *tickets* θ^* or *proof-of-entrance* ν, generated by execution of the system from a V enters a *LEZ* to the V leaves it, the information that connects the messages composing an itinerary is then the certificates $cert_{V_q}$.

Thus, the information that can be used to identify a user $cert_{V_q}$ by an internal (SP) or external attacker (by eavesdropping) is in its certificates. Otherwise, the information of the itinerary does not represent a privacy problem in itself, despite the location and temporal information that it contains. Note that an internal attacker will have all that messages and it will thus reconstruct the whole itinerary. In contrast, an external attacker will have more difficulties with this reconstruction as it will eavesdrop all the communications.

There are two ways of identifying a user by its certificate. On the one hand, it is a certificate extension $cert_{V_q}.idS$ which contains the vehicle identifier V_{id}. However, this information is protected with an asymmetric encryption (in step 2 of Section 3.2.3), using the public key of *PA*. Therefore, this information does not reveal the identity of the user as breaking the RSA encryption is computationally unfeasible nowadays. On the other hand, the certificate can be neither used to identify a user thanks to the fact that the CA, installed in the *SE*, is shared with several users, since the user is registered in an element C of the subset K together with other users in Section 3.2.2. In this way, the issuer of $cert_{V_q}$ cannot be issued as identifier. Note that

in a contrary case, if the size of each element C of K ($num_C = |C|$ was one, vehicles could be identified by their CA.

Assuming that the number of V registered in each element of K (C_i) has an homogeneous distribution and that it is bigger than one ($|C_i| = |C_{i-1}| > 1$, $\forall i \in [1, num_K)$, where $num_K = |K|$), then the probability (*identify*) to link a V with an itinerary, and thus, identify a V, is $\Pr(indentify) = 1/m$ where $m = num_K/num_V = 1/num_V/num_K$ and $num_V = |V|$, that is, the number of Vs that have entered the *LEZ* sharing the same CA.

Hence, the system does not reveal the identity of a V because it cannot be linked to any itinerary (*anonymity*). However, an attacker that is aware of an itinerary will be able to identify a V in a certain probability depending on the number of Vs sharing the same CA that have entered in *LEZ*. □

Claim 13. *The system does not allow to link two itineraries.*

Proof. Since an itinerary is composed of messages or proofs, which place a V in a set of ordered *stretches* such as *tickets* θ^* or *proof-of-entrance* ν, and which are generated by execution of the system from the moment when a V enters a *LEZ* until the moment when the V leaves it, two cases can be considered:

- These two itineraries have associated their two certificates with different CAs. Therefore, it is not possible to link both itineraries because a V cannot change its CA. The probability to link two itineraries, which have associated two certificates with different CAs is then zero.

- These two itineraries have associated their two certificates sharing the same CA. Therefore, it is not possible to associate the two itineraries for two reasons: (i) in the protocol described in step 3.2.3, which is executed each time a V approaches a *LEZ*, the *SE* of each V generates a new $cert_{V_q}$ in each new entrance to the *LEZ*. The certificates used in each itinerary will then be different and they cannot be used to link itineraries; (ii) When two itineraries (with a same CA) can be linked, the first itinerary can be linked to a V and the second itinerary can be linked to the same V. Taking into account that it is not possible that an itinerary cannot be linked with a V as it is stated in Claim 12, this case is neither possible. However, an attacker will be able to link two

itineraries sharing the same CA with a certain probability, which is $1/m^2$, where m is the number of Vs that have entered the *LEZ* sharing the same CA.

Hence, the system avoids the *linkability* between itineraries because the information generated in the system does not link two itineraries or more itineraries. However, an attacker that is aware of several itineraries will be able to link them in a certain probability depending on the number of Vs that have entered the *LEZ* sharing the same CA. □

Claim 14. *The system does not allow to link a payment with a* V.

Proof. Considering that *PS* is a trusted entity, the way in which a V can be linked with a payment done through the messages generated by the *user* and *PS*. These message are *payment-order* ϵ, *transfer-proof-1* ζ_1 and *transfer-proof-2* ζ_2.

payment-order $\epsilon = (ACK, N_C, fing_{PS}, acc_s, amount, acc_d, REF)$ is generated by the user in step 6 of Section 5.6 and signed using its payment credentials. It contains information that can link a payer/user (account source acc_s) and an V (REF). The reference REF is the $hash(\overline{\delta})$, which links with a $\delta = (REQ, N_A, str', N_B, ts, ticket_{id}, fing_{V_q}, amount, \overline{\alpha_{str}})$. As it can observed, δ includes $fing_{V_q}$ and a V could thus be identified. However, this is not possible thanks to the use of a secure communication channel between V and *PS*. Therefore, neither an internal attack nor an external attacker can obtain this information and link an order payment with a V.

PS, after performing a set of verifications, such as the identity of the payer as long as the payer has the necessary funds available, the payment is run and *transfer-proof-1* $\zeta_1 = (amount, acc_d, REF)$ and *transfer-proof-2* $\zeta_2 = (acc_s, amount, acc_d)$ are generates as a receipt in step 7 of Section 5.6. Booth proofs are immediately returned to the user. As in the previous case, both proofs contain information that can be used to link a V with a payment (REF from ζ_1 and acc_s from ζ_2). However, the secure communication channel also protects the disclosure of this information at this point.

V then sends ${\zeta_1}^*$ to *TP* in order to demonstrate the right payment and to obtain a ticket (see step 8 of Section 5.6). After that, every certain period of time and after *PS* sends ${\zeta_2}^*$ to *SP* jointly with other ${\zeta_2}^*$s from other users

and transactions (see step 7vii of Section 5.6). Therefore, *SP* is in possession of a set of ${\zeta_1}^*$ and a set of ${\zeta_2}^*$. In spite of this, *SP* is not able to link ${\zeta_1}^*$ and ${\zeta_2}^*$ since their content does not have information that links them, and the time when *SP* receives both messages is different. However, the *SP* is able to link two ${\zeta_1}^*$ and ${\zeta_2}^*$ from a set of proofs generated in a certain period of time with a probability of $1/l^{*2}$, where l is the number of pairs ${\zeta_1}^*$ and ${\zeta_2}^*$. □

Theorem 5.9.2.6 *The presented system preserves the privacy of honest users and protects their anonymity, thus avoiding the traceability of their actions. Otherwise, the system reveals the identity of dishonest drivers.*

Proof. On the one hand, the presented *ERP* system preserves privacy of honest users in accordance with Claims 12, 13 and 14, being users able to use the system anonymously since each new usage cannot be related to any other with respect to the vehicle identity, and since the payment of a stretch does not link the identity of a vehicle. On the other hand, the system revokes the identity of the dishonest driver when fraud is detected as it is stated in Claim 9. Note that in the case of a fraudulent driver, these drivers will only lose their privacy in the itineraries where they have committed fraud thanks to the fact that they change the certificate as it is stated in Claim 13. □

5.10 Functional Requirements Analysis

This section presents a feasibility study of the application of the proposed system in the real world. This study is divided into two different parts, the *online-feasibility* study and a *traffic simulation*.

The first one, the *online-feasibility* study, aims to evaluate the most critic part of the system and to demonstrate that its execution is possible nowadays. This study involves Sections 3.2.3, 5.7 and 5.6.2, which have hard temporary restrictions since communication between the entities is made in movement. In particular, *V*s, which keep moving in the *LEZ*, Beacons, *Chp*s, *TP* and *PS* have to be able to properly perform each step of these sections

with sufficient time to transmit and receive the expected set of proofs and data (such as certificates, *tickets* or *proof-of-entrance*).

The second study, the *traffic simulation*, attempts to evaluate whether this system is able to effectively distribute the traffic in the *LEZ*. This test has been performed in a simulated environment, using a small topology as an example.

5.10.1 On-line Feasibility Study

This study assesses the feasibility of the practical deployment of some parts of the system. In particular, it is focused on the parts of the system that have high temporary restrictions, where the information transmitted between the entities is in movement. Thus, this study evaluate whether the execution of that parts can be performed quickly enough to have time to transmit and receive the required information in movement. In particular, these parts of the protocol with high temporary restrictions are the *Certificate Generation* phase (Section 3.2.3 of *Certification Protocol*), the *Ticket Acquisition* phase (Section 5.6.2 of *Payment Protocol*), and the whole *Entrance Protocol* (Section 5.7). In the *Certificate Generation*, *V*s have to generated new credentials in a reasonable time previous to the entrance to the LEZ. In the *Ticket Acquisition*, *V*s have to obtain a *ticket* in movement. *Beacon*s, *V*s, *TP* and *PS* entities are involved in this process. Similarly, in the *Entrance Protocol*, *V*s enter a stretch (Section 5.7) and deliver a *ticket* also in movement. *V* and *Chp*s execute this part of the system.

The following development environment and hardware have been used in this study:

- *Mobile Security Card SE 1.0*[3]: This is a Java Card in microSD format, which provides security capabilities for smart phones. Specifically, it uses version 5.0 of the Sm@artCafé Expert's operating system. This device has been used as the *SE* in the simulation because it is a cheap tamper-proof device with cryptographic capabilities.

- *Samsung Galaxy S3*: This smart phone, which uses the Android 4.1.2 version, has been used as an *OBU*. This device has a CPU Quad-core

[3] `http://www.gi-de.com/gd_media/media/press/prs_1/pdf/PM_MicroSD_Card_SE_10_final_E.pdf`

1.4 GHz Cortex-A9, 1 GB RAM and 4G communication. In addition, it has a microSD card slot, which jointly with Android OS, they make communication between smart phones and Mobile Security Card possible. The library Spongy Castle, which is a library for Android platform containing the last version of the cryptographic library Bouncy Castle, is used to perform the less sensitive cryptographic operations out of the *SE*, such as signature and certificate verifications. This device plays the role of *OBU* in the simulation.

- *PC*: It is a common personal computer with a CPU of 2 i7-cores of 2,3 Ghz and 1 GB RAM. The used OS is a 64 bit Ubuntu 14.04. This device will perform the operations of *Chp*s in the simulation. These operations have been developed using the Java language and the cryptographic operations have been computed by using the last version of Bouncy Castle, which is a Java library.

- *Web Service*: Google App Engine, which is a platform as a service cloud computing platform for developing web applications in Java, has been used to carry out the operations to be performed by *PS* and *TP*. The cryptographic operations have also been computed by means of the Bouncy Castle.

The study consists in obtaining the performance cost of the system execution and in evaluating whether this cost is low enough to make communication in movement possible using an affordable hardware and easier to find in the market.

Concerning the communication used, two different communication types are considered. On the one hand, a 4G communication technology has been used in the communication between *V*s, *TP* and *PS* entities. On the other hand, the cost of the communication between *Beacon*s, *V*s and *Chp*s has been estimated from [41], which uses an implementation of ZigBee named XBee[4] and which takes into account the Doppler effect on the communication in movement. The results obtained in [41] prove that the communication in movement between a vehicle at 60km/h and a fixed point with a low number of erroneous packets is possible. In detail, 2000 packets can be correctly

[4] `http://www.digi.com/lp/xbee/`

received, and even more if the velocity of the vehicle is lower. These 2000 packets can be transmitted from 250m of distance, which means that approximately 133 packets per second can be transmitted along it, and that each packet can be fitted with 100 bytes of data.

In view of the aforementioned situation, the total temporal cost has been calculated by implementing the steps of the protocols and by estimating the communication cost, in some cases, and in others, by using 4G. The 4G communication and the computational costs were gathered by averaging the single processes over 100 iterations. The details of the implementation and the performance results obtained in each phase are detailed below:

1. *Certificate Generation* phase (Section 3.2.3): The generation of an entity certificate every time a V is on its way to enter a *LEZ* is implemented. It consists in first generating a RSA key pair (Pk_{V_q},Sk_{V_q}) of 2048 bits in the Mobile Security Card, and then the public key certificate $cert_{V_q}$, following the features described in step 2 (such as the extension containing the encrypted V_{id} with the public key of *PA*). All these have been achieved thanks to the installation of VCA_{C_i} certification entity in the Mobile Security Card. The temporal cost of certificate generation in the Mobile Security Card, which follows the features described in the protocol, is 10,350s, and the size of the generated certificate stored in the smart phone memory is 1574 bytes. Note that the temporal cost covers the key pair generation, the certificate generation and the extraction of it outside the *SE*.

2. *Ticket Acquisition* phase (Section phase 5.6.2): The cost related to the interaction between *V*s and *Beacon*s, *TP* and *PS* to pay and obtain a *ticket* to the *LEZ* is simulated here. The temporal cost of this phase, which is separated according to the computation and communication costs, is detailed below for each step. In addition, the size of the transmitted information in the communication by means of XBee (Step 1) is detailed in order to be estimated.

   - Step 1: Total = 0,032s.
     - Computational cost: 0s.
     - Communication cost: 0,032s (472 bytes = 5 data packets).

- Step 2: Total = 0,9090.
  - Computational cost: 0,8550s.
  - Communication cost: 0,0540s.
- Step 3: Total = 0,3720s.
  - Computational cost: 0,0250s.
  - Communication cost: 0,347s.
- Step 4: Total = 0,0620s.
  - Computational cost: 0,0060s.
  - Communication cost: 0,0560s.
- Step 5: Total = 0,2650s.
  - Computational cost: 0,0050s.
  - Communication cost: 0,2600s.
- Step 6: Total = 0,0830s.
  - Computational cost: 0,0180s.
  - Communication cost: 0,0650s.
- Step 7: Total = 0,2850s.
  - Computational cost: 0,0150s.
  - Communication cost: 0,2700s.
- Step 8: Total = 1,1551s.
  - Computational cost: 1,0571s.
  - Communication cost: 0,0980s.
- Step 9: Total = 0,2620s.
  - Computational cost: 0,0170s.
  - Communication cost: 0,245s.
- Step 9: Total = 0,0020s.
  - Computational cost: 0s.
  - Communication cost:0,0020s.

Thus, the total temporal cost is 3,4273s.

3. *Entrance Protocol* (Section 5.7): The cost related to the interaction between *V*s and *Chp*s to get access to the *LEZ* by sending a *ticket* is simulated here. The temporal cost of this phase, which is separated according to the computation and communication costs, is detailed below for each step. In addition, the size of the transmitted information in the communication by means of XBee is also detailed in order to be estimated.

   - Step 1: Total = 0,118s.
     - Computational cost: 0,008s.
     - Communication cost: 0,110s (1644 bytes = 17 data packets).
   - Step 2: Total = 1,533s.
     - Computational cost: 1,338s.
     - Communication cost: 0,195s (2940 bytes = 30 data packets).
   - Step 3: According to the results of the verifications performed in this step, some of the following sub-steps will be either performed or not. If the verifications are correctly performed (Steps 3i - 3vi, and 3viii), Total = s. Otherwise (Steps 3i - 3vi, and 3vii), Total = s. The detail of the sub-steps is as follows:
     - Steps 3i - 3vi: 0,012s.
       - Computational cost: 0,012s.
       - Communication cost: None.
     - Step 3vii: Total = 0,186s. Note that the time required to obtain the license plate of the V from the camera is taken from [42].
       - Computational cost: 0,011s.
       - License recognition cost: 0,175s.
       - Communication cost: None
     - Step 3viii: Total = 0,023s.
       - Computational cost: 0,004s.
       - Communication cost: 0,019s (267 bytes = 3 data packets).
   - Step 4: Total = 0,001s. Note that this step will be performed if the verifications performed in Steps 3ii - 3vi are correct.
     - Computational cost: 0,001s.

– Communication cost: None.

Supposing that the execution of the protocol is performed correctly, when an incidence has not been generated, that is, the verifications performed in Steps 3ii - 3vi are correct, the total temporal cost is 1,688s. On the contrary, if an incidence has been generated, that is, one of the verifications performed in Steps 3ii - 3vi fails, then the temporal cost is 1,849s.

Therefore, both results are fast enough in order for them to have time to communicate in movement. The temporal length of communication window, in which both entities are able to transmit information at 60 km/h along 250m of distance, is 15s. There is therefore a high margin between 1,849s and 15s, and even more if the velocity of the vehicle is lower.

After these tests, the online feasibility of the system has been stated. In particular, the capability of generating a certificate in a *SE* accomplishing with the protocol within a reasonable period of time, has been proved. Both parts of the protocol with higher temporal restrictions, in which a V obtains a ticket and enters a stretch, *Ticket Acquisition* phase and *Entrance* protocol, have been also demonstrated to be feasible in movement with a reasonable speed. In particular, they take 3,427s and 1,688s, which is a total of 5,115s. This time is equivalent to driving 85,255 m. at 60 km/h. Thus, the length of the payment area should be longer than this value. The signatures performed in the *SE* take almost one second and they become the slowest operations. Further, these results could be easily improved with a more powerful hardware such as a brand new Java Card.

Optionally, the generation cost of a public key certificate $cert_{V_q}$ (10,350s) enables vehicles to change the credentials more frequently than the system considers in order to improve their privacy, provided that the distance between stretches was sufficient to generate the credentials and execute the Ticket Acquisition phase and the Entrance Protocol.

5.10.2 Traffic Simulation

The system presented in this work aims at distributing the traffic in the *LEZ*, and reducing then pollution emissions, by modifying the prices of

each stretch dynamically according to its density. In this section, a traffic simulation, which evaluates the effect of applying the proposed *ERP* system on a traffic network, is presented. This experiment, which is very costly in real environments, is performed with a traffic simulator. The simulation environment and results are detailed in the following subsections.

Simulation Environment:

The simulation environment is characterized by the simulator, the topology (or road map), the traffic demand (routes) and the time duration:

- *Simulator*: The simulator used to perform these tests is the Simulation of Urban MObility (SUMO)[5]. It is a widely used open traffic simulator [48, 49], which is able to model the traffic in the most realistic way. SUMO was developed by the German Aerospace Center (DLR) in 2001, and since then, it has evolved into a set of utilities such as a traffic demand generator and a Dijkstra route planner.

- *Topology*: The traffic network considered is like the one in Figure 5.3. This network is one of the traffic map samples provided by SUMO. In particular, it has the shape of a spider-net with four arms or axes within the net and three circles making up a 24-edge net. The distance between circles is of 100 meters. Each junction of the net has a stop signal.

- *Traffic flow*: The number of trips are one per second throughout the simulation. However, origins and destinations are randomly generated among all nodes in the network. All the vehicle have the same characteristics, that is, same fuel consumption, same pollution, etc.

- *Duration*: The simulation starts in the second zero and ends after 7200 seconds (2 hours).

Simulation Results:

With the aim of proving that the application of the *ERP* system has a positive effect on the traffic, two different tests have been performed with the same

[5] www.dlr.de/ts/sumo/en/

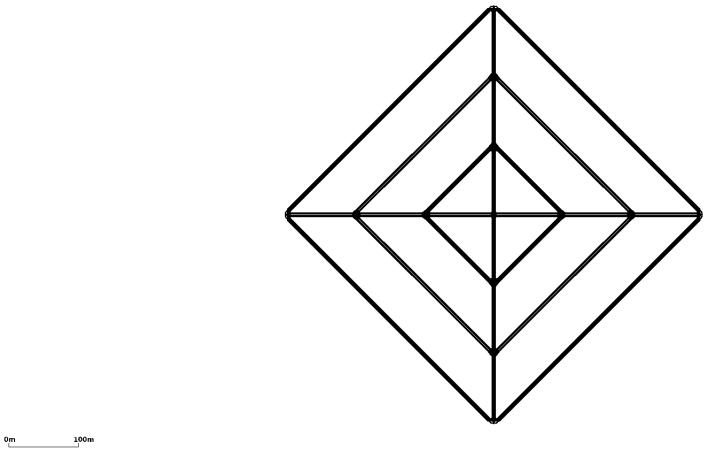


Figure 5.3: Topology

simulator and environment. The first one shows the traffic performance when the users choose the cheapest path taking into account just the distance. In contrast, the second simulation evaluates the traffic performance when the users select the path according not only to the distance, but also to the toll price, which constantly changes according to the vehicle occupancy of the road, that is, in the same way as the proposed *ERP* system.

The evaluation and comparison have been carried out according to a set of metrics. These metrics are:

- *Travel Time:* It is the arithmetic mean of the time needed to reach their destination from their place of origin for all the vehicles in the simulation period.
- *Waiting Time:* It is the arithmetic mean of the time the vehicles are stopped in the network for all the vehicles in the simulation period.
- *Distance:* It is the arithmetic mean of the total distance traveled by all the vehicles to go from their place of origin to their place of destination.
- *Fuel:* It is the arithmetic mean of the fuel consumed by vehicles when traveling through their path.
- *Number of Vehicles:* It is the number of vehicles that entered the network in the simulation period. Note that all vehicles (7200) cannot enter the network due to the network collapse.

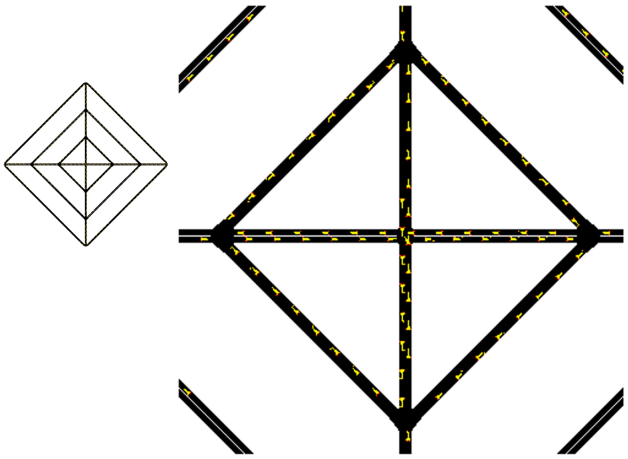


Figure 5.4: Collapse

Both simulations are described and evaluated in detail below.

1. *No ERP system test:* In this case, the traffic has been simulated with static vehicle routes from the mentioned demand. These routes have been computed using the Dijkstra routing algorithm to find the optimal path, taking into account the fuel consumption as edge weight, that is, finding the path with less effort. This effort is the estimated fuel consumption required to travel along the edge, which considers a set of variables such as the edge distance, the maximum speed of the edge, the acceleration of the vehicle or the fuel consumption of the vehicle. The routes have been computed once before starting the simulation. In this way, the vehicles follow their precalculated path during the simulation without changing it.

   The results obtained in the simulation are shown in the second line of Table 5.10.2. As it can be observed, only 574 vehicles entered the network due to traffic jams. The high travel time, 3307,54 seconds, gives an idea of the network collapse, which is too large travel time to cover a few hundred meters. This collapse can be seen in Figure 5.4.

2. *ERP system application test:* Unlike the previous case, the route of each vehicle is dynamically recomputed every second. Although the routing algorithm used here is the same, Dijkstra, the edge weight is not exactly the same. Moreover, these costs change according to the use

of the road, that is, the road capacity, which constantly changes. The formula used to calculate the edge weight is described below:

$$effort_{edge} \times 0,01 + capacity_{edge} \times factor = weight_{edge}$$

The $effort_{edge}$ is the estimated consumption of fuel required to travel along the edge like in the previous case. These units of the effort are *ml*. For this reason, it is multiplied by $0,01$, which is an estimated price per *ml* of fuel (1€/l of fuel). Thus, the first addend is the theoretical cost or effort in Euros if there is no traffic. The second addend is the toll charge, which depends on the number of vehicles that are traveling along the edge multiplied by a factor. The capacity is continuously updated by the system, and it increases when a vehicle enters the edge. The effect of road capacity on travel times is usually specified by means of the so called volume-delay or link travel time functions t(v/q), which express the travel time on a link as a function depending on the link traffic volume v and the link capacity q. One of the most common link travel time formulas is the Bureau of Public Roads (BPR) [50] cost function. Although this function could improve the assessment of the value of the toll addend, for the sake of simplicity, and due to the fact that it is out of the scope of this work, the simulations have been performed with several factor values using the equation above. In particular, the simulation has been carried out with factor values ranging from 0,01 to 2,00 with an increment of 0,001. The max factor supposes a toll addend 127 times higher than the max value of the first addend when the capacity is 20% ($31,413 \times 0,01 = 0,3141$, where $31,413$ is the max effort of an edge).

In order to simplify the results and eliminate the individual effect of the factor values, the arithmetic mean of each metric considering all factor values is showed.

The results obtained in the simulation are shown in the second line of Table 5.10.2. As it can be seen, the results are better than the ones obtained in the "No ERP" case, except for the distance traveled. This is due to the fact that the route of the vehicles tries to avoid roads with high occupancy and then, the vehicles follow longer routes but, still,

Test	Travel Time (s)	Waiting Time (s)	Distance (m)	Fuel (ml)	# of Vehicles
1) No-ERP	3307,54	2849,97	401,15	640,43	574
2) ERP	1869,56	1515,31	637,38	374,99	1687,86

Table 5.3: Results Comparison

get lower fuel consumption. Moreover, the quantity of vehicles that enter the network is larger as there are less traffic jams (the waiting time is small). As a side note, the simulations have shown that a factor equal to 0,494 is the best in terms of travel time for this particular context. This factor reaches mean travel and waiting times of 263,95 and 2,35 seconds, respectively, and mean traveled distance equals to 758,26.

Comparison

In this subsection, after having carried out the simulations, both cases, metric by metric, are compared.

As it can be observed in Table 5.10.2, the results are better in the second case, where dynamic prices are applied for all the metrics except for the distance. Thanks to the change in the edge prices and the frequent route re-planning, the congestion and the travel time are improved. The fuel consumption is also reduced as traffic jams are minimized. The same reasoning can be applied to the number of vehicles, which suggest that with less congestion, more vehicles can enter the net. Finally, It is stated that the distance becomes higher in the second test as the routes of the vehicles avoid congested roads and prefer taking longer routes.

As a conclusion of these simulations, it can be stated that the results applying the *ERP* system on the road network are positive. The comparison has shown that this *ERP* system can reduce the consumed fuel, and thus the air pollution from vehicles, by reducing the travel time and traffic jams. If the comparison is made between the best result obtained with the factor 0,494, the results are even more encouraging. However, in order to determine the the extent to which the use of this system in more realistic networks is profitable, a deeper study needs to be done. This task will be addressed in the future.

Chapter 6

Conclusions

This chapter summarizes the contributions of this thesis in §6.1, the related publications in §6.2, and the possible future research lines in §6.3.

Contents

6.1 Contributions **109**

6.2 Publications **112**

6.3 Future work **112**

6.1 Contributions

The high levels of pollution and traffic congestion that are present in almost all major cities around the world have brought solutions such as the deployment of *Electronic Road Pricing (ERP)* systems in some of these cities. The main goal of these systems is to restrict traffic in the *Low Emission Zone (LEZ)*, for which a toll is assessed according to the traffic and pollution conditions.

These systems have to provide fraud control while drivers' privacy is guaranteed, because poor fraud control does not facilitate the proper operation of the system, and a lack of privacy would result in users being excessively monitored.

The systems proposed in the literature have been analyzed in terms of fraud control and privacy. From this study, it is concluded that they offer probabilistic fraud control that depends on the number of photographs taken of vehicles by checkpoints. Drivers' privacy is then affected since the system records information about their itinerary with more or less accuracy.

For this reason, two new Electronic Road Pricing Systems have been proposed in order to detect fraud in a deterministic way while preserving privacy of honest drivers by means of revocable anonymity.

As it has been stated in this work, group signatures and pseudonym mechanisms do not accomplish with the authentication requirements, which have to be met in a *LEZ*, such as the computational cost and the privacy offered.A new *authentication scheme* based on the RSA signature is thus proposed, which allows users to dynamically generate credentials (public key pairs and certificates) and to sign messages on behalf of the group in a low-cost tamperproof device or secure element with efficient RSA functionalities.

These credentials include a ciphered identifier of the owners of vehicles, which can be used to revoke drivers' anonymity by opening it. In this thesis, it has been proved that the dynamic generation of public key certificates in the secure element using a certification authority, which is installed in it, is feasible, and it has a cost of 10,350s with a key length of 2048 bits. The generation of a public key certificate is performed in a non-critical phase of the protocol. Otherwise, signatures are required in critical phases of the protocol, and so their cost is crucial. The cost of a signature in this device is reasonable since it takes less than one second, which is faster than group signatures [30, 31, 32]. Thus, the resulting scheme provides an anonymous authentication mechanism with revocable anonymity, which has suitable temporal costs for enabling the communication between vehicles and checkpoints in movement. Both proposals use this mechanism in order to preserve users' privacy as long as users collaborate.

The first proposal is based on a time approach where drivers pay according to the time spent in the *LEZ*. In this system, the *LEZ*, as in many real cases, can be composed of a single zone or multiple zones. In the first case, drivers are suggested to use the *LEZ* as less time as possible, or they are just dissuaded from their entrance. In the second case, drivers can be suggested to cover less expensive areas. In both cases, the entrance/departure process of the *LEZ* is controlled so that the legitimate tax is computed while user's anonymity is preserved. Therefore, when a user commits fraud, and only in this case, she is identified according to the way she committed fraud. Two identification methods are provided. In one case, checkpoints identify a vehicle by taking a photograph of the vehicle's license plate. In the other case, Punishment Authority revokes drivers' anonymity by opening the ciphered identifier of their credentials. A security and privacy study has been carried out, which shows that the system offers the required properties. Apart from

this, a study of the practical application of the most critic part of the system has been performed using common hardware such as smart phones and smart cards. Considering the communication and the computational costs, the entrance and the departure processes have a cost of 2,400s and 1,883s, respectively, in the worst case scenario. These results are fast enough to be feasible in movement. This study has also included the evaluation of the feasibility to adapt existing electronic payment systems in order to accomplish with drivers' anonymity and untraceability. In both cases, it has been demonstrated that this proposal is realistic and may be deployed in practical scenarios.

The second proposal follows a dynamic pricing approach independently for each road stretch, where drivers pay according to the paths covered. Unlike the first proposal, the *LEZ*s are defined as a set of stretches. In this case, drivers are suggested to re-plan their route in order to cover less expensive stretches. For this reason, the entrance of each stretch is controlled so that the legitimate tax is dynamically computed depending on the traffic volume while user's anonymity is preserved. As in the first proposal, drivers that collaborate with the system keep their privacy. Otherwise, dishonest drivers are identified through revocable anonymity or registration by Checkpoints. These properties are also analyzed in a study showing that the system offers them. The phases of the protocol where a vehicle enters a stretch and communicates with other entities in movement have been implemented using non custom-made devices.The results obtained from this study show that the costs of obtaining a ticket and then entering a stretch by exchanging it are fast enough, as they take 5,115s. In addition, it has been evaluated whether this system is able to manage traffic by means of a simulation in a small topology. The results are promising and show that the system can reduce the air pollution from vehicles in the simulated environment. However, it requires a deep study in more realistic and complex environments in order to prove it in real environments.

Hence, the last proposal requires a higher infrastructure expense but offers a greater level of impact on traffic by dispersing it from high occupancy stretches. Otherwise, the first proposal implies less infrastructure costs while the effect on the traffic is also lower.

6.2 Publications

The publications supporting the content of this dissertation are stated below:

- Roger Jardí-Cedó, Macià Mut Puigserver, Magdalena Payeras-Capellà, Jordi Castellà-Roca, Alexandre Viejo "Electronic Road Pricing System for Low Emission Zones to Preserve Driver Privacy" In *11th International Conference Modeling Decisions for Artificial Intelligence (MDAI 2014)* pp. 1–13. 2014.
- Roger Jardí-Cedó, Jordi Castellà-Roca, Alexandre Viejo "Privacy-Preserving Electronic Toll System with Dynamic Pricing for Low Emission Zones" In *9th International Workshop on Data Privacy Management (DPM 2014)* pp. 327–334. 2014.
- Roger Jardí-Cedó, Macià Mut Puigserver, Magdalena Payeras-Capellà, Jordi Castellà-Roca, Alexandre Viejo "Time-based Electronic Road Pricing System for Low Emission Zones Preserving Drivers' Privacy" In *International Journal of Computer Communications* 2015. Under review.
- Roger Jardí-Cedó, Jordi Castellà-Roca, Alexandre Viejo "Privacy-Preserving Electronic Road Pricing System for Low Emission Zones with Dynamic Pricing" In *International Journal of Security and Communication Networks* 2015. Under review.
- Roger Jardí-Cedó, Macià Mut Puigserver, Magdalena Payeras-Capellà, Jordi Castellà-Roca, Alexandre Viejo "Sistema de Telepeaje en Zonas Urbanas" In *XIII Reunión Espanola de Criptografía y Seguridad de la Información (RECSI'14). (XIII Spanish Meeting on Cryptography and Information Security)*. pp. 93–99. 2014.
- Roger Jardí-Cedó, Jordi Castellà-Roca, Alexandre Viejo "Urban Electronic Road Pricing Systems Preserving Drivers' Privacy" In *URV Doctoral Workshop in Computer Science and Mathematics (DCSM-2014)*. pp. 23–26. 2014.

6.3 Future work

The work presented in this thesis opens several future research lines:

- In the anonymous authentication scheme proposed, the way in which vehicles are distributed in groups of vehicles that share the same certification authority may affect drivers' privacy. For this reason, a deep study assessing this issue is required. In particular, a test with the traffic simulator used in this work evaluating several compositions of the groups, such as the size of the group or the origin of vehicles, can be useful. In addition, the size of each group has to be studied from the certificate maintenance point of view.

- In both proposals, communication between vehicles and checkpoints has been estimated from another work, which has been put into practice. However, a further study in a more real scenario is required taking into account the interference signals produced by other devices, or just by the coexistence of vehicles entering a checkpoint at the same time.

- In the Functional Requirements Analysis of the second proposal, it is stated that a more detailed study of the impact of applying this *ERP* system on traffic is required in order to determine the extent to which the use of this system in more realistic networks is profitable. In addition, prices of stretches should be fixed taking into account the Bureau of Public Roads cost function in order to obtain the best results.

Bibliography

[1] Raluca Ada Popa, Hari Balakrishnan, and Andrew J. Blumberg. Vpriv: protecting privacy in location-based vehicular services. In *Proceedings of the 18th conference on USENIX security symposium*, SSYM'09, pages 335–350, Berkeley, CA, USA, 2009. USENIX Association.

[2] Xihui Chen, Gabriele Lenzini, Souke Mauw, and Jun Pang. A group signature based electronic toll pricing system. pages 85–93, 2012.

[3] Wiebren Jonge and Bart Jacobs. Formal aspects in security and trust. chapter Privacy-Friendly Electronic Traffic Pricing via Commits, pages 143–161. Springer-Verlag, Berlin, Heidelberg, 2009.

[4] Jaap-Henk Hoepman and George Huitema. Privacy enhanced fraud resistant road pricing. In Jacques Berleur, MagdaDavid Hercheui, and LorenzM. Hilty, editors, *What Kind of Information Society? Governance, Virtuality, Surveillance, Sustainability, Resilience*, volume 328 of *IFIP Advances in Information and Communication Technology*, pages 202–213. Springer Berlin Heidelberg, 2010.

[5] Josep Balasch, Alfredo Rial, Carmela Troncoso, Christophe Geuens, Bart Preneel, and Ingrid Verbauwhede. Pretp: Privacy-preserving electronic toll pricing. In *19TH USENIX SECURITY SYMPOSIUM*, pages 63–78. USENIX Association, 2010.

[6] Sarah Meiklejohn, Keaton Mowery, Stephen Checkoway, and Hovav Shacham. The phantom tollbooth: privacy-preserving electronic toll collection in the presence of driver collusion. In *Proceedings of the 20th USENIX conference on Security*, SEC'11, pages 32–32, Berkeley, CA, USA, 2011. USENIX Association.

[7] Jeremy Day, Yizhou Huang, Edward Knapp, and Ian Goldberg. Spectre: spot-checked private ecash tolling at roadside. In *Proceedings of the 10th annual ACM workshop on Privacy in the electronic society*, WPES '11, pages 61–68, New York, NY, USA, 2011. ACM.

[8] Flavio D. Garcia, Eric R. Verheul, and Bart Jacobs. Cell-based roadpricing. In *EuroPKI*, pages 106–122, 2011.

[9] Flavio D. Garcia, Eric R. Verheul, and Bart Jacobs. Cell-based privacy-friendly roadpricing. *Comput. Math. Appl.*, 65(5):774–785, 2013.

[10] Health Effects Institute. Panel on the Health Effects of Traffic-Related Air Pollution. *Traffic-related air pollution: a critical review of the literature on emissions, exposure, and health effects*. Research report (Health Effects Institute). Health Effects Institute, 2010.

[11] Ying Zhou and Jonathan I Levy. Factors influencing the spatial extent of mobile source air pollution impacts: a meta-analysis. *BMC Public Health*, 7(1):89, 2007.

[12] CJ Peachey, Danielle Sinnett, M Wilkinson, GW Morgan, PH Freer-Smith, and TR Hutchings. Deposition and solubility of airborne metals to four plant species grown at varying distances from two heavily trafficked roads in london. *Environmental Pollution*, 157(8):2291–2299, 2009.

[13] Alex A Karner, Douglas S Eisinger, and Deb A Niemeier. Near-roadway air quality: synthesizing the findings from real-world data. *Environmental science & technology*, 44(14):5334–5344, 2010.

[14] Ulrike Gehring, Joachim Heinrich, Ursula Krämer, Veit Grote, Matthias Hochadel, Dorothea Sugiri, Martin Kraft, Knut Rauchfuss, Hans Georg Eberwein, H-Erich Wichmann, et al. Long-term exposure to ambient air pollution and cardiopulmonary mortality in women. *Epidemiology*, 17(5):545–551, 2006.

[15] Rob Beelen, Gerard Hoek, Piet A van Den Brandt, R Alexandra Goldbohm, Paul Fischer, Leo J Schouten, Michael Jerrett, Edward Hughes, Ben Armstrong, and Bert Brunekreef. Long-term effects of traffic-related air pollution on mortality in a dutch cohort (nlcs-air study). *Environ Health Perspect*, 116(2):196–202, 2008.

[16] Victor C Van Hee, Sara D Adar, Adam A Szpiro, R Graham Barr, David A Bluemke, Ana V Diez Roux, Edward A Gill, Lianne Sheppard, and Joel D Kaufman. Exposure to traffic and left ventricular mass and

function: the multi-ethnic study of atherosclerosis. *American journal of respiratory and critical care medicine*, 179(9):827–834, 2009.

[17] World Health Organization et al. Review of evidence on health aspects of air pollution–revihaap project: final technical report, 2013.

[18] World Health Organization. Regional Office for Europe and World Health Organization. *Air quality guidelines: global update 2005: particulate matter, ozone, nitrogen dioxide, and sulfur dioxide*. World Health Organization, 2006.

[19] BOE. Resolución int/2836/2013. CVE-DOGC-B-14013017-2014. Núm 6541 - 15.1.2014.

[20] RM Qadir, G Abbaszade, Jürgen Schnelle-Kreis, JC Chow, and R Zimmermann. Concentrations and source contributions of particulate organic matter before and after implementation of a low emission zone in munich, germany. *Environmental Pollution*, 175(2):158–167, 2013.

[21] G. Santos. Urban congestion charging: A comparison between london and singapore. *Transport Reviews*, 25(5):511–534, 2005.

[22] Hendrik Wolff. Keep your clunker in the suburb: Low-emission zones and adoption of green vehicles. *The Economic Journal*, pages n/a–n/a, 2014.

[23] F. Costabile and I. Allegrini. A new approach to link transport emissions and air quality: An intelligent transport system based on the control of traffic air pollution. *Environmental Modelling and Software*, 23(3):258–267, 2008.

[24] Vanesa Daza, Josep Domingo-Ferrer, Francesc Sebé, and Alexandre Viejo. Trustworthy privacy-preserving car-generated announcements in vehicular ad hoc networks. *Vehicular Technology, IEEE Transactions on*, 58(4):1876–1886, 2009.

[25] Lei Zhang, Qianhong Wu, Agusti Solanas, and Josep Domingo-Ferrer. A scalable robust authentication protocol for secure vehicular communications. *vehicular Technology, IEEE Transactions on*, 59(4):1606–1617, 2010.

[26] Bo Qin, Qianhong Wu, Josep Domingo-Ferrer, and Lei Zhang. Preserving security and privacy in large-scale vanets. In *Information and Communications Security*, pages 121–135. Springer, 2011.

[27] Lukas Malina, Arnau Vives-Guasch, Jordi Castellà-Roca, Alexandre Viejo, and Jan Hajny. Efficient group signatures for privacy-preserving vehicular networks. *Telecommunication Systems*, 58(4):293–311, 2015.

[28] Maxim Raya and Jean-Pierre Hubaux. The security of vehicular ad hoc networks. In *Proceedings of the 3rd ACM workshop on Security of ad hoc and sensor networks*, pages 11–21. ACM, 2005.

[29] Yipin Sun, Rongxing Lu, Xiaodong Lin, Xuemin Sherman Shen, and Jinshu Su. An efficient pseudonymous authentication scheme with strong privacy preservation for vehicular communications. *vehicular Technology, IEEE Transactions on*, 59(7):3589–3603, 2010.

[30] Andreu Pere Isern-Deyà, M Magdalena Payeras-Capellà, Macià Mut-Puigserver, and Josep L Ferrer-Gomila. Anonymous and fair micropayment scheme with protection against coupon theft. *International Journal of Adaptive, Resilient and Autonomic Systems (IJARAS)*, 4(2):54–71, 2013.

[31] Andreu Pere Isern-Deyà. *Privacy-protecting Systems for Electronic Commerce on Mobile Devices*. PhD thesis, Universitat de les Illes Balears, 2013.

[32] Klaus Potzmader, Johannes Winter, Daniel Hein, Christian Hanser, Peter Teufl, and Liqun Chen. Group signatures on mobile devices: Practical experiences. In *Trust and Trustworthy Computing*, volume 7904 of *Lecture Notes in Computer Science*, pages 47–64. Springer Berlin Heidelberg, 2013.

[33] David Chaum. Security without identification: Transaction systems to make big brother obsolete. *Communications of the ACM*, 28(10):1030–1044, 1985.

[34] Maxim Raya, Panos Papadimitratos, and Jean-Pierre Hubaux. Securing vehicular communications. *IEEE Wireless Communications Magazine, Special Issue on Inter-Vehicular Communications*, 13(LCA-ARTICLE-2006-015):8–15, 2006.

[35] Frederik Armknecht, Andreas Festag, Dirk Westhoff, and Ke Zeng. Cross-layer privacy enhancement and non-repudiation in vehicular communication. In *Communication in Distributed Systems (KiVS), 2007 ITG-GI Conference*, pages 1–12. VDE, 2007.

[36] Adi Shamir. How to share a secret. *Commun. ACM*, 22(11):612–613, 1979.

[37] Mihir Bellare and Phillip Rogaway. Optimal asymmetric encryption. In *EUROCRYPT*, volume 950 of *Lecture Notes in Computer Science*, pages 92–111. Springer, 1995.

[38] Elaine Barker, Miles Smid, Dennis Branstad, and Santosh Chokhani. Sp 800-130. a framework for designing cryptographic key management systems. Technical report, Gaithersburg, MD, United States, 2013.

[39] Elaine B. Barker, William C. Barker, William E. Burr, W. Timothy Polk, and Miles E. Smid. Sp 800-57. recommendation for key management, part 1: General (revision 3). Technical report, Gaithersburg, MD, United States, 2012.

[40] Hua Wang, Xun Yi, Elisa Bertino, and Lili Sun. Protecting outsourced data in cloud computing through access management. *Concurrency and computation: Practice and Experience*, 2014.

[41] Juan Antonio Nazabal, Francisco Falcone, Carlos Fernández-Valdivielso, and Ignacio Raúl Matías. Development of a low mobility ieee 802.15.4 compliant vanet system for urban environments. *Sensors*, 13(6):7065–7078, 2013.

[42] Clemens Arth, F. Limberger, and H. Bischof. Real-time license plate recognition on an embedded dsp-platform. In *Computer Vision and Pattern Recognition, 2007. CVPR '07. IEEE Conference on*, pages 1–8, June 2007.

[43] David Chaum. Blind signatures for untraceable payments. In David Chaum, RonaldL. Rivest, and AlanT. Sherman, editors, *Advances in Cryptology*, pages 199–203. Springer US, 1983.

[44] Satoshi Nakamoto. Bitcoin: A peer-to-peer electronic cash system. *Consulted*, 1(2012):28, 2008.

[45] David L. Chaum. Untraceable electronic mail, return addresses, and digital pseudonyms. *Commun. ACM*, 24(2):84–90, 1981.

[46] Ian Miers, Christina Garman, Matthew Green, and Aviel D. Rubin. Zerocoin: Anonymous distributed e-cash from bitcoin. In *Proceedings of the 2013 IEEE Symposium on Security and Privacy*, SP '13, pages 397–411, Washington, DC, USA, 2013. IEEE Computer Society.

[47] Yasushi Masuda and Seungjin Whang. Dynamic pricing for network service: Equilibrium and stability. *Management Science*, 45(6):857–869, 1999.

[48] Kaveh Shafiee, Jinwoo Brian Lee, Victor C.M. Leung, and Garland Chow. Modeling and simulation of vehicular networks. In *Proceedings of the First ACM International Symposium on Design and Analysis of Intelligent Vehicular Networks and Applications*, DIVANet '11, pages 77–86, New York, NY, USA, 2011. ACM.

[49] Francisco J. Martinez, Chai Keong Toh, Juan-Carlos Cano, Carlos T. Calafate, and Pietro Manzoni. A survey and comparative study of simulators for vehicular ad hoc networks (vanets). *Wireless Communications and Mobile Computing*, 11(7):813–828, 2011.

[50] Urban Planning Division US Department of Commerce. Bureau of public roads. *Traffic Assignment Manual*, 1964.