ARTICLE IN PRESS

Computers and Mathematics with Applications ▮ (▮▮▮▮) ▮▮▮–▮▮▮

Contents lists available at SciVerse ScienceDirect

Computers and Mathematics with Applications

journal homepage: www.elsevier.com/locate/camwa

Cell-based privacy-friendly roadpricing

Flavio D. Garcia [a,*], Eric R. Verheul [a,b], Bart Jacobs [a]

[a] Institute for Computing and Information Sciences, Radboud University Nijmegen. P.O. Box 9010, NL-6500 GL Nijmegen, The Netherlands

[b] Key Controls, Wagnerlaan 33, NL-1411 JD Naarden, The Netherlands

ARTICLE INFO

Keywords:
Privacy enhancing technologies
Pay-as-you-drive roadpricing
Electronic Toll/Traffic Pricing
Electronic parking charging

ABSTRACT

This paper proposes a new approach to electronic roadpricing, based on a division of the roadpricing area into cells, each with their own fee. Some of the cells are secretly marked as check cells. On-board equipment in each vehicle contains a secure element that is made aware of these check cells and helps the pricing authorities to monitor the vehicle's whereabouts in a privacy-friendly manner. This approach is not only original but it also improves upon earlier approaches since it solves issues regarding positioning accuracy, collusion between different users, and the required level of interaction. Moreover, with slight modification this cell-based roadpricing can also be used for automatic handling of parking charges.

1. Introduction

Roadpricing, also known as Electronic Traffic Pricing (ETP), refers to a location-based charge for road use. It exists in several forms, for instance via charges for entering the city center (London) or for particular motorways (France or Italy) or for lorries (Germany). Here we consider satellite-based systems, using GPS or Galileo, for personal vehicles. Such approaches have been elaborated to some extent in the Netherlands, as part of earlier, now abandoned, government plans. However, at the European level this form of roadpricing is still on the political agenda (based on the framework [1,2]). The main reasons for replacing a flat road tax by a location-based approach are: (1) fairness of charges, since one only pays for actual road use, and (2) the possibility to steer the traffic supply via a flexible pricing policy (e.g. making busy roads expensive during rush hours).

Location-based roadpricing for personal vehicles is a highly privacy sensitive matter since it requires detailed location information of individual vehicles. Public support depends on proper privacy protection via an architecture guaranteeing data minimalization focused on the goal of roadpricing itself, and not on secondary goals (like speed limit enforcement). The topic has been picked up in the computer security research community, see [3–6], and several different protocols have been proposed. This paper, which is an extended version of [7], contributes with a novel protocol, which we call cell-based roadpricing (CBR). It uses cells (covering the roadpricing area) both for fee calculation and for fraud detection (via a secure element, like a smart card, embedded in the car's toll device). Our approach addresses some problems with fraud detection in earlier protocols, like time-dependence, collusion, GPS-precision and required level of interaction. The paper does not introduce any new cryptographic primitives and makes use of existing techniques. Its value lies in adapting these techniques in an original manner to a context of considerable societal relevance. Our CBR approach allows certain additional functionality, such as automatic parking-fee collection and a limited form of pay-as-you-drive insurance, see Section 7.

* Corresponding author.
E-mail addresses: flaviog@cs.ru.nl (F.D. Garcia), Eric.Verheul@cs.ru.nl, Eric.Verheul@keycontrols.nl (E.R. Verheul), bart@cs.ru.nl (B. Jacobs).

0898-1221/$ – see front matter © 2012 Elsevier Ltd. All rights reserved.
doi:10.1016/j.camwa.2012.06.022

Please cite this article in press as: F.D. Garcia, et al., Cell-based privacy-friendly roadpricing, Computers and Mathematics with Applications (2012), doi:10.1016/j.camwa.2012.06.022

By 2015 all (new) cars in the European Union will be equipped with the 'eCall' system that automatically contacts Europe's single emergency number 112 in the event of a serious road accident, e.g. a collision or when car occupant pushes an emergency button. The eCall system then communicates the vehicle's location to the emergency services. The eCall system in a car consists of a small on-board unit equipped with GPS and means of communication. We believe that the hardware required for CBR could benefit from the existing eCall hardware as they are similar.[1] By sharing hardware, implementation of CBR could be more cost effective. However, this would need further investigation.

2. Preliminaries

We adopt the main building blocks of the Electronic Toll Pricing (ETP) architecture proposed by the European Union [1]. It distinguishes the following five components/parties of the system:

- an ETP subscriber;
- a toll charger TC;
- a toll service provider TSP;
- an on-board unit OBU in every (personal) vehicle;
- a secure element SE incorporated in the OBU.

An ETP subscriber is the party that uses the road of the toll charger and for this subscribes to a toll service provider. The main information security objectives of the subscriber are twofold. The first objective is that the ETP subscriber gets to pay the correct amount for road usage and that he can validate that. We will call this property *financial integrity* from the perspective of the subscriber. The second objective is that the subscriber's privacy can adequately protected in the scheme. This obviously includes more than protection of the subscriber's road movements but we concentrate on this aspect in this paper. With *adequate protection* we mean that the subscriber accepts the privacy protection design of the toll service provider system and has assurance that the actual implementation adheres to the design ('public trust').

Preferably, the privacy protection the scheme provides should be parameterizable and the subscriber should be able to validate the privacy protection himself without having to rely on a (trusted) third party. The toll charger TC is the party that collects a toll fee for the usage of the roads and defines the prices and conditions of use. Typically national or regional authorities are TCs, but also commercial parties 'owning' the roads, can be a TC.

The toll service provider TSP is the party that provides the ETP service. It periodically determines the fee to be paid by each subscriber (vehicle) for his road usage. In this paper the distinction between TC and TSP is not always clearly made, since it is not so relevant for the protocol. An important information security objective shared by the TC and TSP is that subscribers are correctly charged for the road usage and can detect fraud. A specific security objective of the TC is that it should be able to validate that usage of its roads is correctly reflected in the payments of the TSP. We will call this *financial integrity*, from the perspective of either the TC or the TSP.

The on-board unit OBU is a satellite (GPS or Galileo) enabled device that will be attached to every vehicle subscribed to the ETP service. We assume that this device stores a pricing function $\mathcal{P}$ that takes as input a location l and a time t and outputs a price $p = \mathcal{P}(l, t)$ which corresponds to the toll price of the corresponding road at time t. Furthermore, the OBU must have a timing device which is reasonably in sync with the local time. It should also be able to occasionally communicate with the TSP in order to report usage and to update the pricing function (and a list of checkpoints that corresponds with the location of surveillance cameras, see below). Although the OBU should be reliable it is untrusted, from a systems perspective: subscribers may try to manipulate it, for instance via the power supply or via the satellite signals (shielding the device or feeding it false signals), in order to reduce the fee they need to pay.

The secure element SE is a tamper resistant device, like a smartcard, which has a modest amount of non-volatile memory and processing power. This processing power must be enough to perform basic (public key) cryptographic operations. Typically, the secure element does not have its own power or clock. It is comparable to a SIM card in a mobile phone albeit with more cryptographic capabilities. The private key in the SE determines the cryptographic identity of the SE, and thus of the OBU in which it is embedded.

2.1. Thin and fat OBU's

Before going into the technicalities, we briefly describe the setting and the main idea underlying the protocol that is proposed in this paper.

In all satellite-based roadpricing systems vehicles have an on-board unit OBU that can at least determine the location of the vehicle and communicate with the back-office. In [4] a distinction is made between *thin* and *fat* OBU's. A thin OBU just collects location information and passes it on the back-office where the appropriate fee is calculated. This is a simple but extremely privacy-unfriendly approach, since the sensitive location information is stored outside the direct control of the individual involved, in a large database that is vulnerable in various ways. A fat OBU on the other hand is capable of calculating the price itself, via a pricing function that is executed in a secure environment. At the end of a reporting period the OBU sends the cumulative fee to the TSP.

[1] http://fevr.org/new/2011/09/will-e-call-now-soon-be-introduced.

Please cite this article in press as: F.D. Garcia, et al., Cell-based privacy-friendly roadpricing, Computers and Mathematics with Applications (2012), doi:10.1016/j.camwa.2012.06.022

The fat approach is privacy friendly but has two big disadvantages:

(1) a fat OBU is more complex than a thin one, and thus more expensive and more vulnerable;
(2) fraud detection is more complex with fat OBU's in comparison with thin ones.

One way to handle the first point is to reduce the trusted computing base (TCB) of an OBU to a minimum and to place this TCB on a tamper-resistant secure element SE such as a smart card. This separation makes it easier to add additional (commercial) services to the OBU, or alternatively, to add roadpricing functionality to existing in-car equipment (such as satellite navigation) by adding the secure element.[2]

The second point is a serious issue. First, fraud detection with thin OBU's is easy: a TC/TSP just places roadside cameras at random places and checks if the vehicles that pass by report the location where they are spotted. If not, the car owners will be fined. One sensitive point in this approach is that camera locations must remain secret, throughout the reporting period. If they become known, drivers may simply switch off their OBU and take a different route, avoiding cameras. With a fat OBU one way of fraud detection is to have the TC/TSP communicate briefly from a roadside detection point with the (secure element in the) OBU of each passing car in order to check if the last few locations used by the pricing function are consistent with the check locations. We remark that the ETP architecture [1] actually suggests the usage of 5.8 GHz Dedicated Short-Range Communications (DSRC) for this. However, since such checks are active, involving two-way communication, they will be noticed very quickly by passing cars. The check locations will thus become publicly known, negatively affecting the efficacy of the checks.

Another fraud detection approach is to let the OBU (or its secure element) commit itself to each step of the fee calculation, for instance via a simple hash function [4] or via a more complex non-interactive commitment scheme [3] (using homomorphic encryption and zero-knowledge proofs to exclude negative sub-fees). After receiving a (cumulative) fee report, the TC/TSP may ask the OBU to "open" certain commitments, corresponding to roadside camera locations, in order to check details of the cumulative report. Auditing thus involves interaction with the user, even in the case where everything turns out to be fine. In contrast, with the approach proposed here there will only be interaction with the user in the case fraud is detected.

A crucial problem with the commitment approach is its vulnerability to collusion, exploiting time dependence. During such commitment checks the locations of the road-side cameras become known. If checks are performed immediately, like in [3], people may collude, where one of them (who drives a lot) reports first and learns about the check locations and the other ones subsequently delete commitments for non-check locations in their fee reports (thus reducing their fee).

Hence such checks can only be performed well after the reporting period, when all OBU's have sent in their fee report. But such timing dependency may introduce other vulnerabilities. What should happen if you are on holiday at the end of the reporting period, outside the roadpricing area, and outside reach of the TSP? If you are allowed to send in your report with a delay, when you return, you could adapt your fee report based on knowledge of the checkpoints. Such manipulations can exploit the fact that secure elements do not have their own (secure) clock and so you can feed them false timing information. One way of addressing this "absence" problem is forcing OBU's to report their fee as soon as they leave the roadpricing area. This naturally leads to the idea of carving up the area into cells (where border cells play a special role). Another problem with the commitment approach is that it requires communication with the OBU after fee reporting, allowing fraudsters to claim their OBU is broken or stolen.

We finally note the scheme described in [8]. It has a *thin* OBU in the sense that it is based on a vehicle transponder that constantly sends time-location tuples to the TSP. However these tuples are not associated with vehicles but with commitments to random tags constructed by the subscribers in a registration phase and transferred to their vehicle transponder. During the reconciliation phase the fee is calculated based on a secure multiparty computation involving the TSP and the subscriber through a web application. Enforcement is also based on spot checks; in a zero knowledge fashion the subscriber needs to prove he generated a time-location consistent with the spot check. This scheme not only requires substantial subscriber interaction but also allows the subscriber to claim his tag information is lost.

2.2. Current protocol idea

We now assume that the roadpricing area (country, region, or continent) is covered by square, non-overlapping cells of a relatively small size (e.g. a square kilometer). When you enter a cell you will have to pay a certain amount, depending on the cell (and possibly the time of day). These payment details are incorporated in a payment function $\mathcal{P}$ that will be left unspecified. If you stay longer in the same cell, you may have to pay again, see the discussion in Section 3.2. But more importantly, there will be a new charge as soon as you enter the next cell. In Section 2.3 we elaborate a little more on the organization of these cells.

A certain subset of cells are marked as "check cells" or "checkpoints" and contain (hidden) roadside cameras connected with Automated Number Plate Recognition (ANPR) equipment. These markings are chosen by the TSP and the TC and change every billing period (of, say, one or three months). The secrecy of these markings is crucial for the protocol. Another crucial aspect is that the secure element in the OBU gets to know these checkpoint markings, via a secure connection with the TSP,

[2] The secure element is bound to a particular individual, and can via this separation be transferred from one vehicle to another.

Please cite this article in press as: F.D. Garcia, et al., Cell-based privacy-friendly roadpricing, Computers and Mathematics with Applications (2012), doi:10.1016/j.camwa.2012.06.022

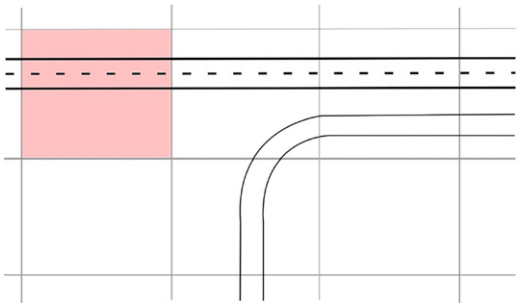

Fig. 1. An expensive cell exclusively covering a motorway segment, in between two exits, surrounded by inexpensive cells.

but the subscriber or its (untrusted) OBU should not learn any information about the current checkpoints. This confidential transfer of checkpoints happens before the beginning of each new billing period. The checkpoint list may be generated in secure hardware and be transmitted via secure channels to the SEs without becoming available in the clear.

Upon entering a new cell, the OBU reports – or 'declares', as we shall say below – this transition to its SE. The SE replies with a "ticket". The ticket is a (randomised) encryption of fixed data, e.g. 'OK' or '0' with some padding if the new cell is *not* a checkpoint. But if the cell is a checkpoint – which is known to the SE but not to the OBU– the ticket will be meaningful for fraud detection later on. The encryptions should be such that one cannot determine which tickets are for checkpoint cells and which are for non-checkpoint cells. In this way the trusted element works against its owner, in the interest of the TC and TSP: it secretly monitors the user. Thus, the SE is the TC and TSP's ambassador (or spy, if you like), see also [9].

Driving around *without* a proper, up-to-date list of check cells is dangerous, because without it your SE cannot produce appropriate tickets when you happen to enter a check cell (leading to subsequent fines, when audited). Hence it is in the own interest of users to have their system up-to-date. Whenever the SE does not have – for whatever reasons – an up-to-date list of check cells, it should report so via an alarm signal to the user. There should be procedures in place for malfunction, for instance by pushing a 'reset' button (including an implicit notification to the TSP) or by going quickly to some service station. Alternatively, one could consider letting the SE resort to treat all cells as check cells when it is not equipped with a proper, up-to-date list of check cells. This implies that the scheme provides no privacy when subscribers do not properly update their OBU.[3]

From a system perspective the secure element is trusted and inaccessible to malicious users. In contrast, the OBU is untrusted. When the trusted element ever gets compromised, the system breaks down. This is usually the case with a TCB: if a SIM gets compromised then identities can be stolen and phone bills will probably end up with the wrong person. But having this tamper resistant SE as TCB also simplifies matters a lot: the SE is trusted so it can simply accumulate the fees per cell, store the result in its non-volatile memory, and report the cumulative fee, appropriately authenticated, to the TSP at the end of the reporting period. Because we have such a clearly separated TCB we do not need to use the homomorphic encryptions and zero knowledge proofs as in [3].

Both the TC and TSP trust the accumulation of fees reported by the SE, but they do not trust that the SE has been notified appropriately by the OBU at every cell transition. The TC and TSP thus check these notifications via the tickets. As part of the fee reporting – or possibly only on request of the TSP– the OBU sends all tickets of the reporting period to the TSP. The TSP can then open the tickets for the checkpoint cells where the vehicle has been spotted and see if the content is appropriate. The TSP can also communicate this evidence to the TC. Details are given below, in Section 3.

2.3. Cells and roads

One thing that is new in our approach is the reliance on cells, and not on road segments or distance, as basis for payment. It deserves some more explanation. One basic idea in roadpricing is that certain roads (e.g. busy motorways) or bridges/tunnels are more expensive than for instance quiet country roads. Simply making all cells in which an expensive motorway occurs expensive is too crude, since there may be multiple roads in a particular cell. What we propose is that the price of a cell with multiple roads corresponds to the price of the cheapest road (in that cell). In order to charge the appropriate fee for an expensive road one cell is made very expensive that only contains this road and cannot be avoided. Such a cell can for instance be found on a stretch of motorway in between two exits, as illustrated in Fig. 1. Of course, such a cell is a prime candidate to be (regularly) a check cell. One (more) advantage of using cells over road segments is that substantial margins can be allowed in the accuracy of the location provided by the positioning system, see Fig. 2. This is a great benefit, since in practice the positioning accuracy is influenced for instance by atmospheric disturbances or by reflections of satellite signals in urban areas. Such lack of accuracy may be a problem for a road-segment-based approach, when there are adjacent, parallel roads with different prices. In contrast, our cell-based approach is more robust. Also, the system is independent of the (often

[3] Jeroen Prins is thanked for this suggestion.

Please cite this article in press as: F.D. Garcia, et al., Cell-based privacy-friendly roadpricing, Computers and Mathematics with Applications (2012), doi:10.1016/j.camwa.2012.06.022

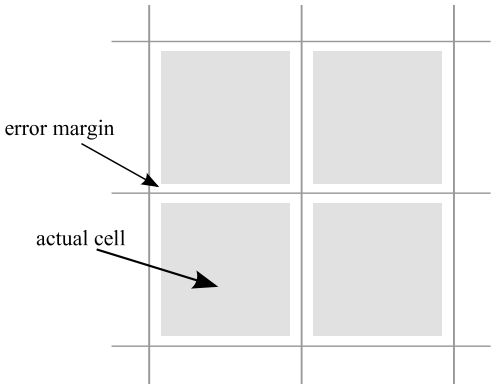

Fig. 2. Positioning system error margin around cells.

proprietary) mechanisms for transforming satellite signals into road segments, on which other roadpricing approaches rely. Hence it can be realized more easily via open (international) standards. Section 6 discusses some issues related to the (ideal) size of cells.

2.4. Timing

As we argued above, the timing of fee reporting, auditing, and feedback is a sensitive matter in road pricing. In our approach, fee reporting takes place:

1. at the end of the reporting period;
2. when a vehicle leaves the road pricing area (e.g. the country);
3. when a vehicle is sold or taken out of service.

The last point is rather special. It can be handled by giving an OBU a special 'report now' button; after pushing it the embedded SE should be removed. The passage of a border will be detected by the OBU and should lead to an (automatic) fee report. It may create some administrative overhead for people who frequently cross borders, but since fee reporting is automatic this should not be such a problem.

It is a more serious problem if the fee report is suppressed by a malicious user upon leaving the road pricing area. In this way a vehicle may disappear from the grid. However, upon re-entering there are two options.

- This re-entering happens 'soon', still within the same reporting period as the departure; then the vehicle can in principle continue to drive around in the roadpricing area, using the (still valid) list of check cells in its SE. The OBU/SE will have to report normally, at the end of this period.
- Re-entering happens 'late', in a subsequent reporting period. There is now a window for fraud, because the check locations from the reporting period of departure must be assumed to be publicly known. However, the vehicle will need an (updated) list of check cells before it can drive around after re-entering without any risk. It is at this point that it has to communicate with the TSP, to request the new list, and the TSP notices the missing report (from the period of departure); the TSP will impose a (hefty) penalty for not reporting upon leaving the territory.

The first, ordinary way of reporting happens at the end of the reporting period, say once a month. Vehicles will be forced to report, because only after reporting will they receive the new list of check cells. Since there is some time in between entering one cell and entering the next one, the fee reporting need not interfere with the ordinary declaring.

After (some time after) receiving the fee reports and distributing the new lists of check cells, the TSP goes into the auditing phase: tickets for check cells will be decrypted and inspected, possibly leading to penalties. Notice that this auditing does not require any interaction with the user — assuming appropriate tickets for check cells are submitted.

(There is one practical issue related to performance. If all vehicles submit their fee report at 24:00 h of the last day of the month, communication channels will be overloaded. Hence one may wish to allocate different reporting times to different vehicles, in order to balance the load. This involves some additional administration, including management of check cells lists and delay of auditing, but it does not change the system fundamentally. More pragmatically one may reserve for monthly fee submission several hours during the night, say between 23:00 and 6:00, and make driving free of charge during that short period.)

3. The protocol

This section describes our new, privacy-friendly ETP protocol. This protocol is a triple $\langle \mathcal{S}, \mathcal{U}, \mathcal{D} \rangle$ of polynomial-time interactive protocols. The Setup Protocol $\mathcal{S}$ is a two-party protocol involving the TSP and the SE. This protocol is run when

Please cite this article in press as: F.D. Garcia, et al., Cell-based privacy-friendly roadpricing, Computers and Mathematics with Applications (2012), doi:10.1016/j.camwa.2012.06.022

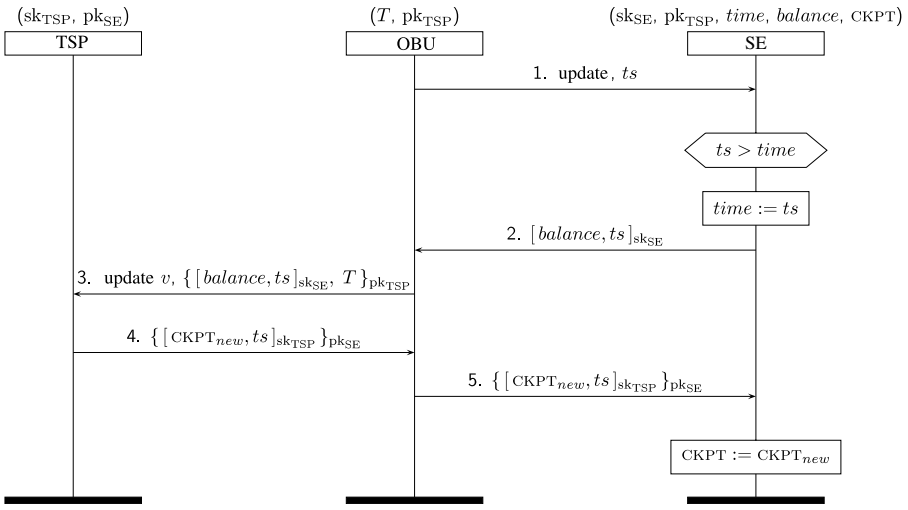

Fig. 3. Update protocol (typeset using the MSC package [10]).

a new SE is added to the system. The Update protocol $\mathcal{U}$ is a three-party protocol involving the SE, the TSP and the OBU. This protocol runs periodically, at the end/begining of a billing period, *e.g.* monthly. The protocol Declare Segment $\mathcal{D}$ is a two-party protocol involving the SE and the OBU. This protocol runs every time that the vehicle enters a new cell — and also when the current cell ticket expires in the case of parking, see Section 7. Furthermore, we assume that the OBU is provided with a clock which keeps track of local time with reasonable accuracy.

Notation. We write $\{m\}_{\mathrm{pk}}$ to denote the encryption of message m with public key pk, while $[m]_{\mathrm{sk}}$ denotes message m followed by a signature on m with private key sk.

3.1. System setup $\mathcal{S}$

When the system is initialized,

- the TSP creates a public key pair ($\mathrm{pk_{TSP}}$, $\mathrm{sk_{TSP}}$) and together with the TC partitions the map of the country (or region under consideration) into small cells, forming a grid that overlaps the relevant area. Let $\mathcal{C}$ be a mapping from GPS coordinates (*lat*, *long*) to a cell number c. These cells must be small enough to provide the desired granularity level for road pricing. The function $\mathcal{P}$ assigns to each cell c and time t the corresponding road price p.
- the OBU stores a ticket list T, which is initialized to the empty list.
- the SE of a vehicle generates its own public key pair (pk, sk). The public key is registered by the TSP and the public key $\mathrm{pk_{TSP}}$ of the TSP is stored in the SE and in the OBU. The SE has a counter *balance*, which is initialized to zero. It also has a register *time*, which represents the SE's local notion of time and can only be increased. This register is updated during a successful Update or Declare protocol execution.

3.2. Update $\mathcal{U}$

The purpose of the update protocol is threefold:

- for the TSP to get the toll charge for the last billing period;
- for the SE to get the updated checkpoint list for the next billing period;
- to perform fraud detection.

At the beginning of each billing period, the TC creates a (secret) list CHECKPOINTS of checkpoint cells. The checkpoint cells are those cells where the TC will have a surveillance camera for the next billing period. These checkpoint cells will be used later on for fraud detection. To prevent abuse from the TC/TSP, the maximum size of the CHECKPOINTS list N_{chk} must be enforced by the SE to be a small fraction of the total number of cells in the map N_{cells}. This prevent TC or TSP to set too many cells in the system as checkpoints, thus hampering privacy. At the end of of each billing period, the TSP, OBU and SE execute the protocol depicted in Fig. 3.

The OBU initiates the update protocol by sending a timestamp *ts* to the SE. The SE checks that the timestamp is in the future with respect to *time* and if so, updates *time* accordingly. In message (2), the SE sends back the current balance *balance* together with the timestamp *ts* signed with its private key $\mathrm{sk_{SE}}$. Next, in step (3), the OBU sends this last message together with the ticket list T, encrypted, to the TSP. In step (4) the TSP sends the checkpoint list for the next billing period

Please cite this article in press as: F.D. Garcia, et al., Cell-based privacy-friendly roadpricing, Computers and Mathematics with Applications (2012), doi:10.1016/j.camwa.2012.06.022

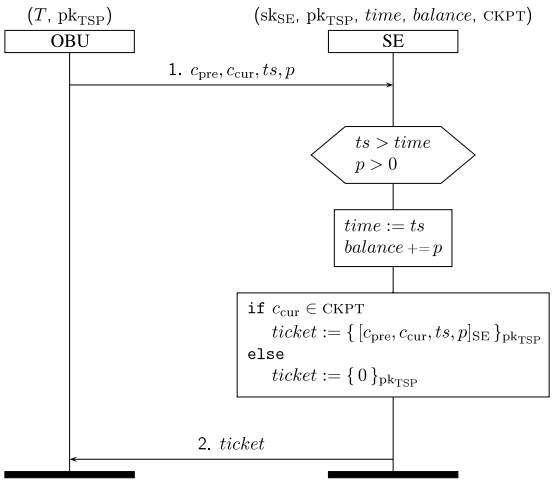

Fig. 4. Declare Segment protocol.

$\textsc{checkpoints}_{new}$ signed together with a timestamp *ts*. This message is encrypted with the public key $\mathrm{pk}_{\mathrm{SE}}$ of the SE. In step (5), the OBU simply forwards this message to the SE. The SE validates that the timestamp *ts* is in the future with respect of its notion of time i.e., $ts > time$ and sets $time = ts$ and updates its checkpoint list.

When the TSP performs fraud detection, it is assumed that the TC provided TSP with evidence that the vehicle v has been at cell c at time t. Then the TSP will decrypt each entry in the transcript list T provided by v (see Section 8 for optimizations), ignoring those entries that decrypt to zero. Entries for (visited) check cells must be of the form $\langle c_{\mathrm{pre}}, c_{\mathrm{cur}}, ts, p\rangle$ with $c = c_{\mathrm{cur}}$ and $ts \approx t$. If that is not the case it means that this vehicle has committed fraud.

The signature over the new checkpoint list on steps (4) and (5) is there to prevent an adversary from being able to send a corrupt (potentially empty) checkpoint list to the SE which would result in fines, or other kind of damage, for the affected vehicles. Note that these signatures are not required for the security of the protocol as defined in Section 4.

To be precise, instead of $ts \approx t$ we should write $t - \delta \leq ts \leq t + \delta$, where δ is a system parameter determining the validity period of a ticket. This constant δ should be chosen carefully, proportional to the cell size. The value of δ must correspond to the maximum time that a vehicle can take to travel through a cell. If δ is chosen too small then slow vehicles will have to pay multiple times for the same cell. Instead, when δ is too large, it allows a vehicle to commit fraud by traveling on a specific route several times while declaring it only once. As rough indication, one can take δ to be 30 min, for 100×100 meter cells, so that a traffic jam does not immediately lead to multiple tickets for the same cell.

3.3. Declare Segment $\mathcal{D}$

Every time that the vehicle enters a new cell c_{cur} the OBU and the SE run the protocol depicted in Fig. 4. The situation for parking is slightly different, see Section 7. In Fig. 4 c_{pre} is the previous cell, $c_{\mathrm{cur}} = \mathcal{C}(lat, long)$ is the current cell, *ts* is the local time and $p = \mathcal{P}(c_{\mathrm{cur}}, ts)$ is the toll price. Upon receipt of message (1), the secure element verifies that the cell c_{pre} corresponds to the c_{cur} of the previous Declare Segment protocol run. The SE validates that the timestamp *ts* is in the future with respect of its notion of time *i.e.*, $ts > time$ and checks that $p > 0$. Then the SE adapts its local time, by setting $time = ts$, and increases the counter *balance* by *p*, and subsequently it returns a ticket (using randomised encryption). If the cell $c_{\mathrm{cur}} \in \textsc{checkpoints}$ then the SE sets $ticket := \{[c_{\mathrm{pre}}, c_{\mathrm{cur}}, ts, p]_{\mathrm{SE}}\}_{\mathrm{pk}_{\mathrm{TSP}}}$. Otherwise it sets $ticket := \{0\}_{\mathrm{pk}_{\mathrm{TSP}}}$. The OBU stores the ticket from message (2) in the ticket list T, until the next update protocol execution.

We remark that the subscriber's OBU can, independently of the SE, store all declarations in plaintext and keep its own balance register, allowing the subscriber to validate the charges sent to the TSP during the update protocol. We also remark that one can easily let the cell numbers support a notion of adjacency, allowing the SE to determine that the previous and current cell are indeed adjacent allowing an additional form of fraud detection.

4. Security notions

This section introduces different security notions. Most of it is standard and is taken from the literature. It first recalls the notion of indistinguishability under chosen ciphertext attacks for an encryption scheme and the notion of strong existential unforgeability under chosen message attacks for a signature scheme. Then we propose a new security definition for ETP protocols that capture realistic threats that have not been considered in previous security definitions from the literature.

Definition 4.1. An *encryption scheme* is a triple $\Gamma = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ of probabilistic polynomial-time algorithms. $\mathcal{K}$ takes as input the security parameter 1^{η} and produces a key pair (pk, sk), where pk is the public encryption key and sk is the private

Please cite this article in press as: F.D. Garcia, et al., Cell-based privacy-friendly roadpricing, Computers and Mathematics with Applications (2012), doi:10.1016/j.camwa.2012.06.022

decryption key. $\mathcal{E}$ takes as input a public key pk and a plaintext m and outputs a ciphertext. $\mathcal{D}$ takes as input a private key sk and a ciphertext and outputs a plaintext or $\perp$. It is required that $\mathbb{P}[(\text{pk}, \text{sk}) \leftarrow \mathcal{K}(1^\eta); c \leftarrow \mathcal{E}(\text{pk}, m); m' \leftarrow \mathcal{D}(\text{sk}, c) : m = m'] = 1$. We write $\{m\}_{\text{pk}}$ to denote $\mathcal{E}(\text{pk}, m)$.

Definition 4.2 (*IND-CCA-Game*)**.**

$$\begin{array}{l}\textbf{IND-CCA-Game}_{\Gamma,\mathcal{A}}(\eta)\ :\\ \quad (\text{sk}, \text{pk}) \leftarrow \mathcal{K}(1^\eta)\\ \quad p_0, p_1 \leftarrow \mathcal{A}_0^{\mathcal{D}}(\text{pk})\\ \quad b \leftarrow \{0, 1\}\\ \quad b' \leftarrow \mathcal{A}_1^{\mathcal{D}}(\{p_b\}_{\text{pk}})\\ \quad \textbf{winif if } b = b'.\end{array}$$

Adversaries implicitly pass state *i.e.*, from $\mathcal{A}_0$ to $\mathcal{A}_1$.

Definition 4.3 (*IND-CCA*)**.** An encryption scheme Γ is said to be *IND-CCA* secure if for all probabilistic polynomial-time adversaries $\mathcal{A} = (\mathcal{A}_0, \mathcal{A}_1)$

$$\mathbb{P}[\textbf{IND-CCA-Game}_{\Gamma,\mathcal{A}}(\eta)] - 1/2$$

is a negligible function of η. This adversary has access to a decryption oracle $\mathcal{D}$ that on input a bitstring c' outputs $\mathcal{D}(\text{sk}, c)$ with the only restriction that c is not equal to the challenge ciphertext $\{p_b\}_{\text{pk}}$.

We recall the definition of signature scheme and the notion of strong existential unforgeability under chosen message attacks.

Definition 4.4. A *signature scheme* is a triple (Gen, Sign, Vrfy) of probabilistic polynomial-time algorithms. Gen takes as input the security parameter 1^η and produces a key pair (vk, sk), where vk is the signature verification key and sk is the secret signing key. Sign takes as input sk and a message m and produces a signature s of m. Vrfy takes as input vk, a message m and a signature s and outputs whether or not s is a valid signature of m.

$$\begin{array}{l}\textbf{SEU}_{\Sigma}(A):\\ (\text{vk}, \text{sk}) \leftarrow \text{Gen}(1^\eta)\\ m, \sigma \leftarrow A^{\$_{\text{sk}}}(\text{vk})\\ \textbf{winif } \text{Vrfy}(\text{vk}, (m, \sigma))\end{array}$$

We recall the standard notion of strong existential unforgeability under chosen message attacks [11].

Definition 4.5 (*Strong Existential Unforgeability*)**.** A signature scheme $\Sigma = (\text{Gen}, \text{Sign}, \text{Vrfy})$ is *strong, existential unforgeable* if the success probability of any probabilistic polynomial-time adversary A in the game $\textbf{SEU}_{\Sigma,\mathcal{A}}(\eta)$ is negligible in the security parameter η.

Next we introduce a new security notion for ETP protocols. Here an adversary is able to navigate a number of vehicles at will and it is also able to communicate with a number of secure elements and with the TSP. The adversary wins the game if it is able to get a toll fee reduction for any vehicle and manages to do so with low probability of being detected. Low probability here means lower probability than that of simply driving by through a random cell without declaring it. Such an adversary models, for instance, the situation where a number of users collude in order to avoid toll charges. This threat has not been considered by Balasch et al. [3].

Definition 4.6 (*Security-Game*)**.**

$$\begin{array}{l}\textbf{Security-Game}_{\Pi,\mathcal{A}}(\eta)\ :\\ \quad \text{params} \leftarrow \text{Setup}(1^\eta)\\ \quad v \leftarrow \mathcal{A}^{\mathcal{O},\mathcal{D}}(\text{params})\\ \quad (\vec{l}, \vec{t}) \leftarrow \text{Trajectory}(v)\\ \quad p, b \leftarrow \text{Result}(v)\\ \quad \textbf{winif if } b = 1 \text{ and } p < \mathcal{P}((\vec{l}, \vec{t}))\end{array}$$

where the adversary $\mathcal{A}$ has access to an oracle $\mathcal{D}$ which on input a vehicle identifier v and a GPS coordinate g, models the (physical) movement of vehicle v from the current location to GPS coordinates g. If g corresponds to a cell $c \in$ CHECKPOINTS then the oracle provides the corresponding evidence to the TSP. $\mathcal{A}$ has also access to an oracle $\mathcal{O}$ which allows her to communicate to the other parties (in this case the SE and the TSP) in the ETP protocol Π. The function Trajectory, on input a vehicle identifier v returns a vector of pairs Location $\times$ Time, corresponding to the trajectory of vehicle v that has been submitted to the oracle $\mathcal{D}$ during the previous billing period. The function Result, on input a vehicle identifier v returns the toll charge p which v has to pay as a result of the last Update protocol run, together with a bit b which equals one when no fraud has been detected for vehicle v. The function $\mathcal{P}$ is overloaded in the definition to vectors of Location $\times$ Time pairs and computes the correct price of this trajectory.

Please cite this article in press as: F.D. Garcia, et al., Cell-based privacy-friendly roadpricing, Computers and Mathematics with Applications (2012), doi:10.1016/j.camwa.2012.06.022

Definition 4.7 (*Security*)**.** An ETP protocol $\Pi = \langle \mathcal{S}, \mathcal{U}, \mathcal{D} \rangle$ is said to be *secure* if for all probabilistic polynomial-time adversaries $\mathcal{A}$

$$\mathbb{P}[\textbf{Security-Game}_{\Pi,\mathcal{A}}(\eta)] < N_{chk}/N_{cells} + negl(\eta)$$

where *negl* is a negligible function on its input η.

5. Security and privacy of the protocol

Theorem 5.1 (*Security*)**.** *If the encryption scheme Γ is IND-CCA secure and the signature scheme Σ is strong, existential unforgeable then the protocol proposed in Section 3 is secure with respect to Definition 4.7.*

Proof. Assume by the contrary that there is an adversary $\mathcal{A}$ that wins the **Security-Game** with probability significantly larger than N_{chk}/N_{cells}. In order for $\mathcal{A}$ to be able to win the Security-Game, it must be able to drive a vehicle through a cell without declaring it or do so but with a lower price. In order to win with probability significantly higher than N_{chk}/N_{cells}, it must either be able to construct valid tickets for the checkpoint cells or be able to avoid checkpoint cells with non-negligible advantage. In the first case case we build the following adversary $\mathcal{B}$ against the IND-CCA security of the encryption scheme. In the second we build the following adversary $\mathcal{C}$ against the strong unforgeability of the signature scheme.

The adversary $\mathcal{B}$ will first simulate the environment for $\mathcal{A}$, that is, it will proceed as in System Setup, creating all public key pairs by itself, except for pk_{SE} of a randomly chosen vehicle v. For this key $\mathcal{B}$ will use the challenge key from the IND-CCA game to $\mathcal{A}$. Then, $\mathcal{B}$ creates a checkpoint list containing one single cell c, sets the challenge plaintext $p_0 = \{c\}$ and sets p_1 equal to the empty checkpoint list. $\mathcal{B}$ receives then a challenge ciphertext $\{p_b\}_{\text{pk}_{\text{SE}}}$ which it uses in step 4 of the Update protocol for v. At some point $\mathcal{A}$ stops and outputs a vehicle identifier v'. If $v \neq v'$ then $\mathcal{B}$ stops and outputs a random bit b'. Otherwise $\mathcal{B}$ picks a random cell c' and compares how many times the vehicle v drives through c and c'. If v drives through c' more than through c then $\mathcal{B}$ stops and outputs zero. On the other hand, if v drives through c more (or equally) often than through c' then it outputs one.

Next we construct an adversary $\mathcal{C}$ against the strong unforgeability of the signature scheme. Just as before, $\mathcal{C}$ will simulate the environment for $\mathcal{A}$, creating all public key pairs by herself, except for pk_{SE} of a randomly chosen vehicle v. For this key $\mathcal{C}$ will use the challenge key from the SEU game. Whenever $\mathcal{C}$ needs a signature using sk_{SE} it will simply invoke the signing oracle $\mathcal{S}_{\text{sk}}$. At some point $\mathcal{A}$ stops and outputs a vehicle identifier v'. If $v \neq v'$ then $\mathcal{C}$ stops as well and outputs $\perp$. If $v = v'$ then $\mathcal{C}$ will analyze the transcript of the Update protocol for vehicle v submitted by $\mathcal{A}$ to the oracle $\mathcal{O}$, in particular to message 5. $\mathcal{C}$ decrypts this message (since it has possession of sk_{TSP}) and all tickets in the ticket list T. As a result $\mathcal{C}$ obtains a number of signatures (ignoring those tickets that decrypt to zero). From these signatures $\mathcal{C}$ ignores those that it has submitted to the signing oracle $\mathcal{S}_{\text{sk}}$. From the remaining signatures it chooses one at random and stops. In the case that there is no such a signature, then it outputs $\perp$. □

Theorem 5.2 (*Privacy*)**.** *If the checkpoint list matches the actual positioning of surveillance cameras, then the scheme proposed in Section 3 does not leak any extra location information about the vehicles.*

Proof. It is straightforward to see that whenever the current position of a vehicle is not a checkpoint cell, then the only information received by the TSP is an encryption of zero. □

6. The size of cells

An interesting question is what the ideal size of cells is. As we outlined in Section 2 the ETP architecture needs to find a balance between privacy on the one hand and financial integrity on the other. The privacy sensitive event at stake is when a subscriber's position becomes known to the system. There are two occasions when this may happen:

1. when the subscriber's car is photographed by a road-side camera; necessarily, the cell in which this camera is located is a check cell;
2. when the subscriber passes through a check cell and reports this passage — assuming that the subscriber follows the protocol.

Ideally, these two events coincide: every passage through a check cell leads to a photo registration. If not, there is a loss, either of privacy, or of efficiency of checks. The unjustified loss of privacy conflicts with the proportionality principle in Article 6 of the European directive on the protection of personal data [12]. This means that cells should be relatively small, because with larger cells the chance is higher that one visits a check cell without being photographed. The question remains: how small should cells be? We shall argue below that from an auditing perspective they should not be very small.

Now, when we look at the financial integrity aspect we need to look at the CBR system from an audit perspective. For the auditor the system is just like any other system where sale slips are involved, and where the check cell tickets play the role of sale slips he can validate. To provide revenue assurance the auditor will typically take a percentage of sale slips (and thus of check cell tickets) and validate if these are correctly handled, i.e. if the check cells tickets are consistent with the photos taken. Compare [13].

If we assume that all cells generate roughly the same number of tickets, we conclude that the percentage of tickets the auditor requires corresponds with the percentage of check cells. Note that irrespective of the size of the cells, this

Please cite this article in press as: F.D. Garcia, et al., Cell-based privacy-friendly roadpricing, Computers and Mathematics with Applications (2012), doi:10.1016/j.camwa.2012.06.022

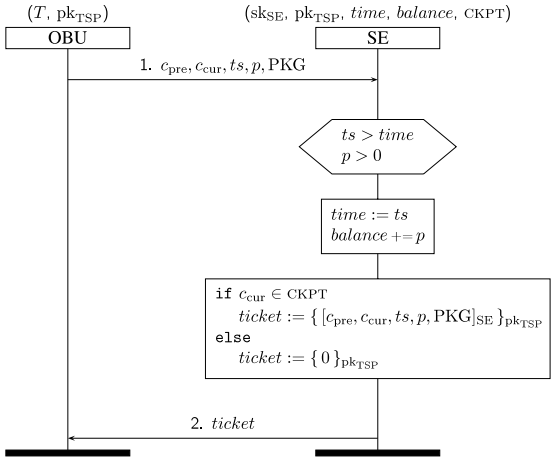

Fig. 5. Declare Protocol with parking.

percentage corresponds both to the probability of detecting fraud and to the percentage of revenue reviewed by the auditor. To summarize, from the financial integrity objective it follows that a certain, fixed percentage of cells need to be check cells, independent of the size of the cells.

However, under this condition the *total number* of check cell tickets generated is influenced by the size of the cells. Indeed, when the cells are small then in absolute terms there will be many check cells and consequently there will be many check cell tickets and thus many events where the subscriber loses privacy sensitive data. So under the assumptions stipulated above it seems that large cells are best for privacy. In addition, large cells seem best for the system from a technical reliability perspective. Indeed, the errors in position systems will have the highest influence in small cells. Here a worst case scenario would be that these errors would make the OBU believe that the subscriber is moving between cells where in fact he is standing still.

7. Extensions

We briefly discuss two possible extensions of the CBR system, for parking and for pay-as-you-drive insurance.

7.1. Parking

In a straightforward manner, the system can be modified to support parking fee payment. This can be achieved by simply adding a status bit PKG to the Declare protocol from Section 3, representing whether the vehicle is in "driving" or "parking" state, see Fig. 5.

The transition from one status to another can be done manually by having the user push a button on the OBU; or it can also be done automatically, e.g., the OBU automatically changes to "parking" state when the user removes the key from the vehicle's ignition. When the vehicle starts moving again, the system goes back to "driving" status.

In parking mode, tickets have possibly a different validity time δ' than in driving mode. This δ' corresponds to the minimum payable parking time, typically five minutes. After this time period, the parking ticket will expire and then the OBU needs to initiate another Declare protocol instance to get a new ticket. As before, if the parking cell is in the checkpoint list then the corresponding ticket will be submitted to the TSP, allowing for inspection. The parking fee is aggregated by the SE in the balance register or in a different register used exclusively for parking.

Parking requires additional fraud detection, via cameras that occasionally drive through a parking area that is marked as check cell, in order to take photos. Such mobile checks are already happening in some existing parking schemes. Possibly, additional check cells have to be added.

7.2. Pay-as-you-drive insurance

With road pricing the flat road tax is replaced by a fairer system based on actual, personal usage. In the car insurance business the same movement is visible, where a flat insurance fee is being replaced by a fee that depends on the distance/time/location, or also on behavioral aspects like driving style. This approach also requires that cars be equipped with boxes, monitoring certain characteristics. In such "pay-as-you-drive" schemes privacy is a serious concern, and privacy-friendly approaches are appearing [14].

The CBR approach described in this paper can support pay-as-you-drive insurance, but only to a limited extent. Of course, the OBU can register various driving characteristics and pass them on directly to an insurance company, but the interesting

Please cite this article in press as: F.D. Garcia, et al., Cell-based privacy-friendly roadpricing, Computers and Mathematics with Applications (2012), doi:10.1016/j.camwa.2012.06.022

question is if the secure element SE in the OBU can play a role in a privacy-friendly scheme. A separate insurance pricing function can be added that translates cells plus time-of-day into an insurance cost. The SE can keep track of the resulting cumulative amount, and produce tickets for each cell update. For fraud detection the insurance company will need a contract with the toll service provider TSP in order to use the same photo registrations in check cells.

8. Practical considerations and extensions

This section elaborates on some practical issues concerning getting public trust in the SE, regarding memory, communication and computational complexity of the protocol and it discusses possible optimizations.

With respect to getting public trust in the SE we suggest that it is certified against an appropriate common criteria profile and that its source code (e.g. JavaCard applet) is disclosed. In addition, we suggest that after a certain period of time the checkpoint lists are also disclosed and that subscribers can get access to cryptographic keys through the TSP allowing them to inspect the actual tickets sent in the update protocol. To further convince subscribers that they are not sent checkpoints specifically tailored for them one could send the checkpoints in through broadcast (e.g., through the Radio Data System (RDS) which is commonly used for dissemination of road traffic information). Using RDS would also allow for regular checkpoint updates without the need for the subscriber to be connected to two-way communication networks such as GSM, GPRS or UMTS that also allow pinpointing the subscriber's location. Actually, it might also be cheaper to use RDS broadcasts than point-to-point communication.

To avoid usage of a static cryptographic key shared by all SEs in this context, we suggest that a common temporary session key is negotiated with all (non-revoked) SEs as part of the update protocol.

If we consider cells of 100 by 100 m, then in a European context 32 bit cell numbers will be sufficient. The protocol described in Section 3 generates a ticket list of approximately 1 MB per 1000 km. Given that the average car usage in Europe is on the order of 15000 km per year, this accounts for 15 MB of data per car per year, which is manageable but can be improved.

It is possible to reduce the computational and bit complexity of the protocol by a small modification in the Declare Segment protocol $\mathcal{D}$. This modification consists of accumulating a number of tickets at the SE and issuing only one ticket every N cells. This ticket contains the same information as the earlier N tickets, but it requires only one signature and one encryption. This modification does not affect the security of the scheme as long as the resulting ticket is of constant size, i.e., padding should be used for those tickets that are just zero. For $N = 10$ this optimization reduces the size of the transcript file to just 200 KB per 1000 km, which accounts to roughly 3 Mb of data per year for an average user. If we consider monthly reports, the transcript file that needs to be sent will be on average of size 300 KB.

If this modest communication complexity is considered unacceptable, it can further be reduced by auditing only a small random set of users. In Step (3) in the Update protocol $\mathcal{U}$ the vehicle can first commit to a certain balance and then the TSP decides probabilistically whether it will request the ticket list T or not. This comes of course as a trade-off for security but it can be a viable option when the sanctions for fraud are severe enough.

Both optimizations reduce the amount of data that needs to be transmitted and the computational complexity of the protocol. This is also important for the TSP. In order to search for the ticket for a particular spot check, the TSP needs to perform a number of decryptions. The timestamp in the tickets can be used to search through the ticket list in (probabilistic) logarithmic time.

We remark that the scheme can obviously support more than one balance register corresponding with cell cost categories instead of actual monetary values. In this way the OBU does not need to be updated with actual cost information; the transformation from cell cost categories to actual cost can be done by the TSP. We also remark that the scheme trivially supports toll roads by just letting toll road segments correspond to (expensive) cells. Furthermore, the scheme might employ several checkpoint lists, which allows more flexible update periods and preventing overload of the communication network. Of course, roadside cameras need to be aware of which car belongs to which checkpoint list in order to prevent unnecessary monitoring. This can be achieved by partitioning the licence plates among checklists, e.g., vehicles with licence plate ending on x (or some cryptographic variation) are assigned checkpoint list x.

9. Conclusions

We have proposed a novel protocol for Electronic Traffic Pricing. This protocol has fairly relaxed accuracy requirements for positioning hardware, making it easy to implement in practice, in an open manner. The protocol is resilient to colluding adversaries where the positioning of the checkpoints remain secret, even after the reporting of road usage to the TSP. In contrast to other systems from the literature, this reporting procedure does not require user interaction, which would drastically improve its acceptability. Additionally, the current approach only involves interaction with the user when fraud is detected, and not during honest usage.

Acknowledgments

We wish to thank Gert Maneschijn, corporate security officer of the Dutch vehicle authority RDW for many stimulating discussions on road-pricing and its security and privacy concerns. The present scheme can be considered as an enhancement of a scheme developed by him and Eric Verheul in 2008.

Please cite this article in press as: F.D. Garcia, et al., Cell-based privacy-friendly roadpricing, Computers and Mathematics with Applications (2012), doi:10.1016/j.camwa.2012.06.022

References

[1] Directive 200452EC of the European Parliament and of the Council of 29 April 2004 on the interoperability of electronic road toll systems in the Community, 2004.
[2] Commission decision of 6 october 2009 on the definition of the European Electronic Toll Service and its technical elements, 2009.
[3] Josep Balasch, Alfredo Rial, Carmela Troncoso, Bart Preneel, Ingrid Verbauwhede, PrETP: Privacy-preserving electronic toll pricing, in: USENIX Security, 2010.
[4] Wiebren de Jonge, Bart Jacobs, Privacy-friendly electronic traffic pricing via commits, in: P. Degano, J. Guttman, F. Martinelli (Eds.), Formal Aspects in Security and Trust, in: Lecture Notes in Computer Science, vol. 5491, Springer, Berlin, 2009, pp. 143–161.
[5] Jaap-Henk Hoepman, George Huitema, Privacy enhanced fraud resistant road pricing, in: Jacques Berleur, Magda Hercheui, Lorenz Hilty (Eds.), What Kind of Information Society? Governance, Virtuality, Surveillance, Sustainability, Resilience, in: IFIP Advances in Information and Communication Technology, vol. 328, Springer, 2010, pp. 202–213.
[6] Xihui Chen, Gabriele Lenzini, Sjouke Mauw, Jun Pang, A group signature based electronic toll pricing system, in: 2nd Grande Region Security and Reliability Day, 2011.
[7] Flavio D. Garcia, Eric R. Verheul, Bart Jacobs, Cell-based roadpricing, in: S. Petkova-Kikova, A. Pashalidis, G. Pernul (Eds.), 8th European PKI Workshop, EuroPKI 201, in: Lecture Notes in Computer Science, vol. 7163, Springer, Heidelberg, 2012, pp. 106–122.
[8] Raluca Ada Popa, Hari Balakrishnan, Andrew Blumberg, VPriv: Protecting Privacy in Location-Based Vehicular Services, in: 18th USENIX Security Symposium, Montreal, Canada, August 2009.
[9] David Chaum, Torben P. Pedersen, Wallet databases with observers, in: E.F. Brickell (Ed.), Advances in Cryptology: Proceedings of Crypto'92, Springer-Verlag, London, 1985, 89–105.
[10] Sjouke Mauw, Victor Bos, Drawing message sequence charts with LaTeX, TUGBoat 22 (1–2) (2001) 87–92.
[11] Shafi Goldwasser, Silvio Micali, Ronald L. Rivest, A digital signature scheme secure against adaptive chosen-message attacks, SIAM Journal on Computing 17 (1988) 281–308.
[12] Directive 199546EC of the European Parliament and of the Council of 24 october 1995, 1995.
[13] Information Systems Audit and Control Association. Is auditing guideline. IT Governance, 2008.
[14] Carmela Troncoso, George Danezis, Eleni Kosta, Joseph Balasch, Bart Preneel, PriPAYD: Privacy-friendly pay-as-you-drive insurance, IEEE Transactions on Dependable and Secure Computing 8 (5) (2011) 742–755.

Please cite this article in press as: F.D. Garcia, et al., Cell-based privacy-friendly roadpricing, Computers and Mathematics with Applications (2012), doi:10.1016/j.camwa.2012.06.022