RUHR-UNIVERSITÄT BOCHUM

Horst Görtz Institute for IT Security

Technical Report TR-HGI-2013-001

Improving Location Privacy for the Electric Vehicle Masses

Tilman Frosch[*], *Sven Schäge*[†], *Martin Goll*[*], *Thorsten Holz*[*]

[*]Chair for Systems Security, [†]University College London

Ruhr-Universität Bochum
Horst Görtz Institute for IT Security
D-44780 Bochum, Germany

TR-HGI-2013-001
June 6, 2013

RUHR UNIVERSITÄT BOCHUM RUB

hgi Horst Görtz Institut für IT-Sicherheit

Improving Location Privacy for the Electric Vehicle Masses

Tilman Frosch*, Sven Schäge†, Martin Goll*, Thorsten Holz*

Abstract

Electric vehicles (EVs) are becoming increasingly popular, especially since we need alternatives to cars powered by fuel. One main characteristic of EVs is that conventional gas stations become superfluous: since even a quick charging cycle of an EV takes about 30 minutes for a full charge today, we need a more flexible way to charge EVs. As a result, networks with many thousands of so called charging stations (CS) are being built, where a car owner can plug in her car and charge it. The worrying side-effect of this change in how we charge cars is that suddenly this process becomes observable: while today everyone can buy fuel at a gas station in an anonymous way, e-mobility (and especially the billing process) changes the rules significantly and enables an observer to track where a user charges her car.

Simply replacing cash with e-cash would solve most privacy problems in this context. Correctly applied, e-cash can offer a strong protection for customers' privacy, but lack comparable incentives for the vendor to use it.If vendors should endorse a certain solution, it needs to be beneficial (or at least acceptable) to both sides.

In this paper, we tackle this challenge and propose a system that balances the customer's legitimate interest to preserve her location privacy with the vendor's legitimate interest to prevent abuse and the legal requirement to be able to resolve disputes in front of a court of law. The system also supports to authenticate a user in a non-repudiable way in compliance with pre- and post-payed billing such that billing can be handled correctly. Our approach is based on a group signature scheme that we adapt to the setting of next-generation cars. To study the practical feasibility of the proposed system, we implemented a prototype and evaluate it both on a CS for EVs and also on a (simulated) backend. The evaluation results suggest that our system can process more than one million charging processes per hour using off-the-shelf hardware while enabling location privacy.

1 Introduction

Primer on Electric Vehicles In a world where oil-dependent mobility comes increasingly under scrutiny, electric vehicles (EVs) are one possible alternative (again). While around the year 1900 about 38% of the cars in the US were powered electrically [1], in the following decades the EV had been marginalized by cars with internal combustion engines. Today, electric vehicles are once again a promising concept for solving some of the environmental and transport challenges we are facing as a civilization. In modern times, EVs have been used by enthusiasts for local transport for the better part of the last thirty years. Today, many major car manufacturers offer a series of EVs or plan to do so in the next one or two years. By December 2012, about 53,000 plug-in hybrid (PHEV) or battery electric vehicles (BEV) have been sold in the US [2] and market research predicts up to 3.4 million annual world-wide sales of PHEV and BEV in 2020 [3]. EVs are also increasingly considered as mean of state- or even nation-wide transport. For a customer, this scenario often boils down to the question "Where can I charge my vehicle, when I am not at home and will I get there?", also referred to as *range anxiety*. Today in both North America and Europe, there are more than ten thousand charging stations (CS) accessible to the public: PlugShare.com [4] lists about 11,000 charging stations in the US and Canada, while for example LEMnet [5] lists more than 4,000 charge points throughout Europe and neither service is (or claims to be) in possession of an exhaustive list. For the near future, the European Commission proposes a minimum target of 795,000 publicly accessible charging stations throughout the EU and a total number of 7.96 million charging stations [6].

In terms of nation-wide transport, the US car manufacturer Tesla started to roll out so called quick charge stations along transit roads in California and has plans to do throughout the US. However, these charging stations are only accessible for customers of Tesla's latest premium line of vehicles [7]. In Europe, small countries like the Netherlands take a different approach: *stichting e-laad*, a non-profit foundation backed by government and energy utilities, has so far set up more than 2,300 public charging stations throughout the country [8]. Customers have to authenticate themselves before they can charge, which is often done via an RFID card. If, for example, a customer in the US wants to use charging stations of major providers like ChargePoint [9], Blink [10], and SemaCharge [11], he will need contracts and means of authentication for each system. In this growing market, the number of companies providing charging stations to the public is likely to increase. From the customer's perspective, this calls for a system that enables him to use any charging station based on one contract with one entity, and without the need to carry more than one authentication token.

Additionally, the need for a clearing process arises, as customers of different utilities charge at the same charging station. However, both scenarios require utilities to communicate the energy consumption of individual customers amongst themselves. While this might work well for a small number of associated partners, companies will face scalability issues once the number increases. Following the example of the banking and telecommunications sector, at least two parallel efforts are already under way in the energy sector to establish a clearing house to back a roaming-enabled charging infrastructure for electric vehicles.

Privacy Problems When refuelling a conventional vehicle, the customer normally has the option to pay cash at the gas station, thus not linking her identity to her location by means of a financial transaction. In an electric mobility scenario, this is rarely possible. While unattended gas stations that only accept credit cards also exist, this scenario is not comparable to an e-mobility scenario: if a gas station does not accept cash, a customer that desires to use this payment method can simply avoid this station. If there is to be a network of interoperable CSs that is convenient and transparently to use for the customer, payment methods need to be uniform for every CS. Thus, the customer does not have the same freedom of choice.

For the foreseeable future, charging an electric vehicle will take much longer than refuelling a traditional car. One implication of this circumstance is that a customer will not drive to a specific place for refuelling. Instead, recharging will take place at the location the vehicle is anyway (e.g., in a public parking lot or a parking lot near work). As a result, utility companies are gearing up to replace oil companies as mobility providers. In this process, they expand their traditional billing scenario to their new field of business. In most parts of the world this is either a subscription-based (post-paid) model or a scenario where the customer deposits a certain amount of money in her account with the power supply company and is eligible for a given amount of energy (pre-paid). The common element in both scenarios is that the customer's identity is linked to each payment process. Cost benefits of pre-existing billing infrastructure aside, there exist other reasons not to offer cash as a billing option, primarily that each charging station (or group of stations) would need to retain a certain amount of cash. Maintaining a low cash level in all stations does not scale well in a widely distributed infrastructure, while retaining a larger amount of cash in each charging station solicits theft or at least vandalism.

Still, we want to maintain that privacy, and, in the given scenario, especially location privacy is desirable. Blumberg and Eckersley define location privacy as "the ability of an individual to move in public space with the expectation that under normal circumstances their location will not be systematically and secretly recorded for later use" [12]. A vehicle's movement profile allows to infer habitual behavior of its owner, both correctly and incorrectly so. For example, if a person regularily charges her EV in front of a rehabilitation clinic, an entity interpreting available location data might (falsely) suspect a history of drug abuse. A person who charges her car in a red-light district on a regular basis may want to keep this information to herself [13]. Questionable use of location data is not far-fetched: records of Integrated Transportation Payment Systems (ITPS), like E-Z Pass and Fast Lane, have already been used by divorce lawyers to prove that, suspectedly

cheating, husbands have not been where they claimed to be at a given time [14]. The inability to preserve the customers' location privacy may also hamper business interests: the lower adaption rate of FasTrak in the Bay Area, as compared to similar ITPS in comparable urban areas of the United States, is partially related to consumers' value of perceived privacy overweighting the value of convenience [15]. While a user may choose to have her movement profile available (which can for example be desirable for fleet management), the safe default in any charging, billing, and clearing infrastructure should be that the customer's location privacy is preserved.

Privacy-preserving payment schemes offer a well-researched solution to this problem, i.e., in many setups, where cash payments would be desirable, cash could be replaced with e-cash. Regrettably, this excellent solution is suddenly off the agenda, when the vendors' side, i.e., the utilities, are unwilling to accept anonymous customers. By proposing the implementation of anonymous payments to utilities, we learned that anonymous payments more often than not are not an option. While we respect the legitimate business interests of the utilities and acknowledge that anonymous payments might be undesirable for some, we are convinced that a customer's legitimate desire for location privacy, too, must be protected. Instead of indirectly protecting the customer's location privacy by obscuring or hiding the customer's identity, we propose and implement a solution that directly protects the customer from disclosure of her location, but allows for the identity of the customer to be known to the vendor, effectively preventing the creation of traces of spatio-temporal locations.

However, we believe that the binding of billing data to individuals also offers several advantages to customers and utilities that cannot be achieved when integrating e-cash. This immediately follows from the fact that the average volume of used electric energy will be much higher for people with electric vehicles than for those using traditional cars. This also means that EV owners buy more electricity and thus may obtain higher discounts when combining their general electricity needs with those arising from their EVs. To this end, customers may either exploit (standard) rate options where the customers commit to a certain interval of usage per month suited to their overall needs – excesses might be relative expensive while the base rate is cheap – or entirely unique rate options specifically crafted for customers with EVs (or low-emission vehicles), for example [16]. It would be favorable, technically feasible, and reasonable to also have customers be fined for these fees outside of the private garage (i.e., at charging stations), remotely similar to mobile communication networks where each customer can communicate throughout the country while being charged with a unique rate. Of course there may also be options where it is cheaper or more expensive to charge within a certain area or within a certain time of the day, depending on the costs incurred to produce electricity at that point in time. We believe that this would greatly increase the acceptance of customers (specifically commuters). From the utilities' point of view this may increase the binding of their customers to them and allows to better adapt the production of electricity to their customers' needs.

Contributions In this paper, we propose a system that allows for the charging of electric vehicles, authenticates a user in a non-repudiable way in compliance with pre- and post-payed billing, and preserves the user's location privacy. More precisely, we make the following contributions:

- We employ a carefully chosen group signature scheme with strong security properties that provides very efficient verification procedures for large numbers of signatures as a central building block of our system. We adapt this scheme to also allow for full compliance with regulations. The privacy mechanisms protecting the user's location data are very strong: not only is it impossible to decide whether a user has charged her vehicle at a specific CS, it is even impossible to decide whether a user has *ever* been charging at one or several CSs more than once. In compliance with local laws, the system still allows a trusted third party to revoke the location anonymity of past billing processes in cases of a dispute.
- Our solution is complete in the sense that it covers every step necessary from authenticating the customer prior to a charging process to providing all information necessary for the clearing process and does not require new structures, but closely fits existing clearing and billing structures while it can be implemented efficiently on a large-scale.

- To the best of our knowledge, we are the first to offer an implementation of a practical charging and billing system for electric vehicles that offers strong protection of the customer's location privacy. Our implementation performs well even on the limited hardware of a CS, while we are able to process more than one million charging processes per hour using off-the-shelf hardware at the backend.

2 Overview

The system we propose for improving location privacy during the charging process of EVs consists of three main phases: authenticating the customer, authenticating the tuple of customer identity and energy consumption data, and transmitting this data to a clearing house, all without compromising the customer's location privacy. We employ a group signature scheme and adapt it to the specific needs of our setting. In the following, we first describe the attacker model before presenting our scheme.

2.1 Threat Model

Throughout this paper, we assume that an attacker is able to observe and actively interfere with data transmitted on the network. An attacker is able to observe traffic patterns either at a given set of CSs or at the backend. In the given scenario, one entity does not operate both, as CSs are operated by electric energy utilities and the backend is operated by the clearing house. We assume that the entity able to perform the opening process that reveals the signer of a group-signed message is part of the organizational structure of the entity operating the backend. Thus nothing can be gained for this entity by collaborating with an attacker able to observe traffic patterns at a charging station. We assume that the opener is honest and acts lawfully. However, an attacker may have access to the network that connects the backend or a CS to the Internet. Such an attacker might be able to infer the origin of a message by use of a timing side channel, but be unable to attribute the connection to a specific user, as the user's identity is transmitted confidentially.

We assume that an attacker may have complete control over at least part of the network of CSs, but that the attacker is unable to alter the CSs themselves. The latter assumption is sound, as for energy law regards each charging station as a point of sale that has to be audited, calibrated, and its correct function certified with respect to all relevant use cases. While modifications of a CS may be subtle and go undetected, we consider these kind of attacks to be out of scope of this publication.

2.2 User Authentication

Only minor parts of electrical energy is consumed where it is produced. Thus, utilities deliver electrical energy to the grid for distribution. In an e-mobility scenario, charging stations are the end points of the grid, and points of interaction with the customer. Before the customer can connect her EV to a CS, she has to authenticate herself towards the station. Today, some infrastructure providers aim for or already implement an online solution, where the customer presents credentials that the CS forwards to a backend. Upon approval, the backend sends a command to the CS to unlock the plug and/or enable the energy flow. The *Open Charge Point Protocol* (OCPP) [17], an open industry standard currently used in several European countries, also follows this pattern. An obvious disadvantage of this procedure is that once a CS is offline, no customer can charge, meaning both the customer is probably stuck (because she relied on this CS to reach a certain destination) and the utility company losing revenue.

We aim at providing a solution that allows for offline authentication of a customer with the help of a public key infrastructure (PKI). JavaCard Open Platform 41 v.2.3.1 (JCOP 41 v2.3.1) embedded smartcard controllers support RSA [18] for en-/decryption, signature generation and verification, and random number generation [19] and thus can serve as an authentication token

in a PKI. The authentication process can be performed offline, as a challenge-response protocol between the smartcard and the CS, thus temporary unavailability of the CS's internet connection is not an issue in our approach.

2.3 Privacy-preserving Payment vs. Privacy-preserving Authentication of Metering Data

Due to technical and legal issues, many utilities just bill their electromotive customers by the hour. For this to change, proper binding of metering data and authenticated customer identification data is required that will hold up to legal scrutiny. From a consumer's perspective, it is highly desirable that she is not only billed correctly according to her energy consumption, but also her privacy is adequately protected. Several anonymous payment system have been proposed in the past (e.g. [20–23]), discussed controversially and also been accused of enabling or easing criminal behavior, like money laundering or blackmailing [24]. Still, privacy-preserving payment schemes are a very good solution in this scenario and can adequately protect the customer's location privacy, if applied correctly. They allow for the detection of double-spending as soon as a coin is redeemed at the bank and thus allow to detect abuse. To further protect utilities from fraudulent customers, mechanisms to selectively block anonymous users could be applied. Such mechanisms have been proposed for instance by Johnson et al. [25] and Tsang et al. [26] to prevent abuse in anonymity networks and to exclude repeatedly misbehaving users.

However, if the vendor, i.e., the utilities are unwilling to accept anonymous customers and/or untraceable payments there exists a high probability that no privacy-preserving payment scheme will find its way to this market.

Instead of proposing another anonymous payment system, we try to balance the customer's legitimate interest to preserve her location privacy with the vendor's legitimate interest to prevent abuse and the legal requirement to be able to resolve disputes in front of a court of law. We aim to protect the customer's location privacy although each charging process can be attributed to a customer account. Also, the payment process itself is out of scope of this paper. While we are in no way opposed to energy utilities handing out pre-paid authentication tokens to anonymous customers, we aim at preserving a customer's location privacy, even if the authentication token is linked to the customer. Furthermore, authentication tokens not bound to a customer can nevertheless act as a pseudonym for the customer, thus making her transactions linkable and enabling the creation of movement profiles. Under the premise that the customer must be identifiable, our work deals with the question of whether it is still possible to build systems where customers enjoy strong forms of location privacy. Conceptually, we thus must deviate from the widespread paradigm of anonymizing customers in privacy-enhancing systems. Our new approach to this problem is to *anonymize locations*, i.e., charging stations.

One way to cryptographically bind a customer identity to metering data resulting from a charging process are *Message Authentication Codes* (MAC) to ensure the integrity of the information without anyone being able to tell which party generated the MAC. However, a MAC does only prove that some party in possession of the symmetric key has created a message, it does not offer non-repudiation. Thus, successful dispute resolution between customer and utility company (or clearing house) is hard, as typically each party that is able to verify the correctness of a message is also able to generate a correct message.

Digital signature algorithms achieve non-repudiation. However, they are not only applied to guarantee the authenticity of the signed data, but also to authenticate entities. By providing a digital signature on a fresh message, a communication partner shows that it possesses the secret key that corresponds to the public key under which the signature verifies successfully. In this sense, classical signatures are bound to the asymmetric keys of their creators in a unique way. In our case this would mean that the identity of the CS (and therefore the customer's location) is implicitly known, as it signed the message so the backend (BE) at the clearing house is able to verify that the message has not been altered.

While strong sender-anonymity is the main additional security property required in our system, we have to consider that in many countries energy laws or standard weights and measures laws

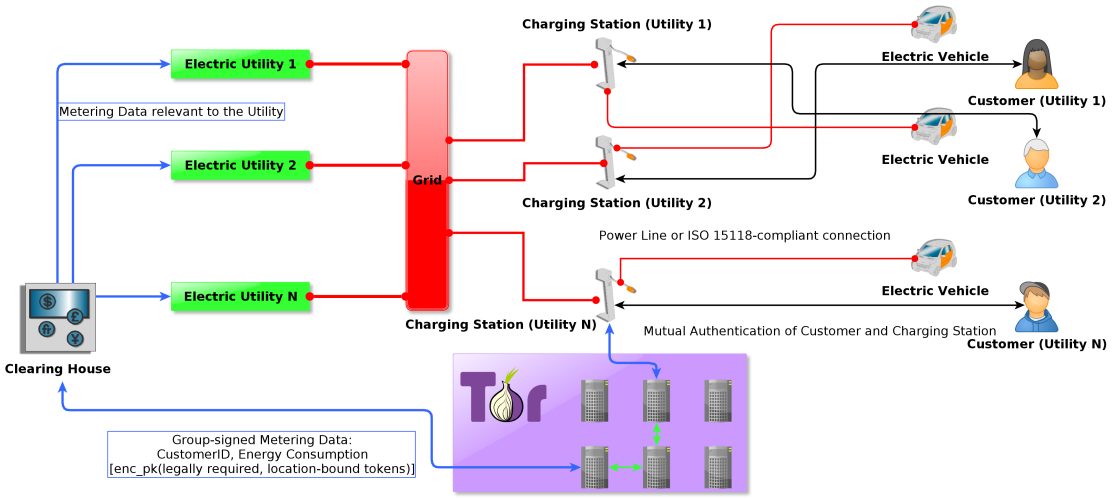

Figure 1: Charging and Transmission of Metering Data

require more data for lawful dispute resolution. For example, if a customer complains about a bill item, it might be required by law, as well as, desirable for the customer, to identify the energy meter that produced the respective consumption data. Thus, it is desirable to use a signature scheme that allows for the *conditional* identification of a signer, while in the default case allowing her to remain anonymous. If such a scheme is used, a high legal hurdle must be placed before the identification of a signer (i.e., the respective CS). This could mean, for example, that a court order or the customers consent is required. Group signature schemes that support an opening mechanism match these requirements very well as detailed below. For every entity that is not in possession of the opening key, the actual signer of a message is indistinguishable from every other potential signer within the same group. Thus, while a customer's transaction is always linked to his customer account, our system guarantees unlinkability with respect to location and time of a transaction.

In summary, the authentication and charging process we propose is as follows (cf. Figure 1):

1. A customer initiates the authentication process by holding up her smartcard to the reader in front of the CS.

2. CS authenticates towards smartcard and vice versa. The CS retains the authenticated customer identity.

3. Upon successful authentication, the CS's power outlet is unlocked and/or put on-line. Charging begins as soon as the EV is connected.

4. When the power-line connection between the CS and the EV is interrupted, the CS generates a tuple containing all information required for the billing process, i.e., the authenticated customer identity stored from Step 2, the amount of energy provided to the user, a timestamp indicating the beginning of the charging process and a timestamp indicating its end. Each location-bound token that is legally required is encrypted to the single entity in possession of the opening key. The tuple is signed using the group secret x_i of the respective CS and the data is transmitted to the clearing house.

2.4 Location Privacy-preserving Transmission of Metering Data

So far, we proposed a protection mechanism for the user's location privacy on the application layer by using a group signature scheme to ensure the integrity of billing relevant data. However, by transmitting the data tuple directly from the CS to the clearing house we would compromise the

user's privacy, as the data available from the network layer would allow for the identification of the CS (e.g., by its IP address). Thus, we utilize a low-latency mix network to transmit the data. The probably most well established implementation of such a mixnet is the Tor network [27]. To ensure confidentiality of the transmitted data, we establish a TLS tunnel between CS and backend prior to transmission.

As the clearing house aggregates and verifies metering data from all the CSs, it is capable to provide either only data clearing or also financial clearing to the associated electric utility companies, which in turn allows each utility's customers to roam freely between all other utilities cooperating with the clearing house.

2.5 Group Signatures and XSGS

Group signature schemes are an essential part of our approach and thus we explain in the following how we utilize and adapt this concept. The idea of group signature schemes has first been introduced by Chaum and van Hejst in 1991 [28]. A group signature scheme is a digital signature scheme that (additionally) provides a (strong) form of sender-anonymity. Unlike in classical signature schemes where each signature is produced by a single signer, in a group signature scheme each signature is produced on behalf of a group. For the verifier it is easy to check whether the signature has been produced by one of the current *group members*. However finding out who exactly produced the group signature is impossible. Intuitively, the larger the group is, the better are the anonymity guarantees provided for each group member – an ideal property for our scenario.

Anonymity: Pseudonyms vs. Group Signatures Group signatures provide a very strong form of anonymity that is usually referred to as *unlinkability*: it is not only impossible to map a signature to its creator – this could be achieved by pseudonyms alone. Unlinkability also implies that noone, except for a dedicated trusted party called *opener*, is able to decide whether two group signatures have been produced by the same signer. We believe that for our application this property is crucial[1]. When using pseudonyms for CSs alone to protect the user's location privacy, the verifier could easily build up customer profiles for every CS which, with more and more user-dependent billing data, could possibly be narrowed down to a single CS. In this way one could easily reveal the true CSs behind the pseudonyms. As a consequence, the verifier could easily follow where and when each user charged its vehicle. Group signatures on the other hand do not even reveal whether two signature belong to the same CS. So users who constantly charge their vehicle at the same CS are indistinguishable from those who travel a lot and often use CSs that they have never visited before.

Group Manager and Opener Technically speaking, a group signature is not verified against a single user's public key UK, but using a group public key GPK that is generated, held, and updated by a trusted third party (TTP) called the *group manager* or *issuer*. The group manager is similar to a classical certification authority. It issues credentials – a more complex form of certificates that also hide the users identity – and updates information on which credentials have been revoked, for example due to adversarial corruptions of the secret key.

Besides the issuer, there is another TTP called the *opener*, who behaves like a notary. Unlike the group manager, the opener has no equivalent in classical PKIs. It holds a secret key which (by calling an opening algorithm) allows to revoke the anonymity of a group signature and provide a proof (which is publicly verifiable) of who the actual creator of a group signature is. The opener can be called in cases of dispute where the identity of the signer of a group signature is of prime importance. For example, if a CSs has been compromised, the adversary may try to continously generate faked bills of its competitor to incur high electricty costs. In these cases, a law court could ask the opener to reveal the identity of the compromised CS. This CS could then i) immediately be revoked by the group manager to avoid further attacks and ii) be analysed, fixed, and equipped

[1]We recall once again that user identitities have to be known to the verifier for a proper billing process. Thus it is not possible to anonymize user identities in the bills.

with fresh key material by a maintainance service. Also the opener could help to find out which other parties might have been attacked in the same way.

Signature Generation and Verification On a technical level, a group signature differs considerably from a classical signature. Intuitively, each user U holds a credential c, which consist of several values that constitute a signature with respect to the group public key GPK. Among these values there is a commitment C for which only U knows the corresponding decommitment. This decommitment constitutes the user's secret key UK, and must be held secret by U. When U joins the group, the group manager only signs the commitment C instead of UK. To convince the group manager that U (the prospective group member) actually knows UK, U has to also deliver a proof of possession of UK. Now, a group signature consist of the following: a) a probabilistic encryption Z of c encrypted with the openers public key and b) a message-dependent, non-interactive proof that i) U knows valid c and corresponding UK with respect to GPK and ii) that the exact same c is encrypted in Z to the openers public key. The proof is an non-interactive zero-knowledge (NIZK) proof which reveals no information beyond the validity of the statements [29,30]. It shows that U is a valid, registered user in the group defined by GPK and that whenever the opener desires it can reveal the identity of the signer of the group signature as U using its secret opening key. For verification, the receiver of a group signature simply checks the NIZK proof with respect to the group public key GPK. On sucess it accepts the signature as valid, otherwise it rejects.

Design Features of the XSGS Scheme Group signatures vary in the extent of functionality they offer and in the security guarantees they provide for group members and verifiers. In our work, we utilize the *eXtremely Short Group Signature* (XSGS) scheme by Delerablee and Pointcheval [31]. The XSGS scheme is an extended variant of the well-known group signature scheme by Boneh, Boyen and Shacham (BBS) which achieves very high efficiency with respect to both signature size and speed [32]. It modifies the BBS scheme in two ways. First, it adds improved protection of group members against collusions of (corrupted) members who try to frame a user. In XSGS, even if the issuer itself is corrupted and takes part in that collusion, its honest group members cannot be framed. Second, XSGS guarantees unlinkability of signatures to even hold against an adversary that can convince the opener to open all other signatures. BBS does in general not cover such attacks (not even when the adversary may convince the opener only once). As a theoretical benefit of these extensions, the XSGS scheme can be proven secure in the very strong security model of Bellare, Shi, and Zhang [33]. We believe that these extended properties of XSGS are practically also neccessary in our application. In particular, they allow to implement the issuer at the same place as the (only) verifier (i.e., the clearing house), without risking the CS's anonymity.

Support for Batch Verification An important design restriction of our solution is that we consider a single verifier that has to verify a huge amount of signatures. The group members on the other hand, do only have to generate a moderate amount of signatures each day. Thus our group signature scheme should ideally feature very fast verification procedures. Kim et al. showed that XSGS supports batch verification [34]. In general, batch verification [35, 36] identifies the most expensive operation in the verification of a signature scheme and combines the verification procecedures of several signatures into a single one such that this operation is only executed for a (small) constant number of times (which is independent of the number of signatures). This can greatly improve efficiency. For security, the combination process is setup in such a way that adversaries cannot produce a combination of invalid signatures which pass the batch verification test[2]. In the XSGS verification, the most expensive operation is the evaluation of a bilinear operation (the so-called pairing) executed on elements of certain elliptic curves. This operation is usually applied in each signature verification. The batch verifier for the verification process only requires the pairing to be called once. Batch verifiers pay off when the (expected) number of invalid signatures per batch is very small. In theses cases one can easily apply a divide-and-conquer approach to identify the invalid signatures. One recusively divides the batch into two

[2]The batch verifier of Kim et al. uses the so-called small exponent test [36].

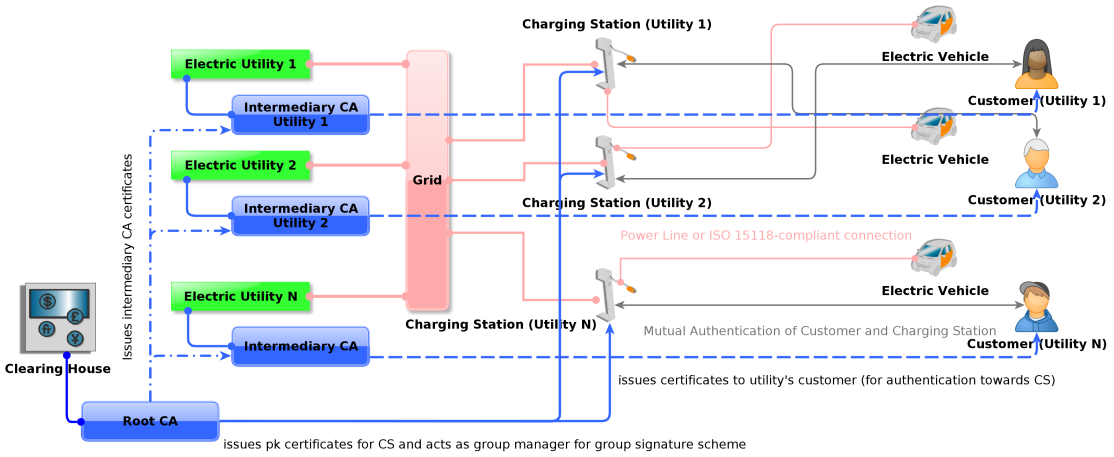

Figure 2: PKI for User Authentication

halfes and applies batch verification to these halfes. Every batch that does not successfully pass the batch verification is again divided up into two new batches of half the size. This process incurs at most $\log_2(n)$ (where n is the size of the original batch) sub-processes per invalid signature. However, for larger number of invalid signatures, this process quickly performs inferior to the separate verification of each signature.

Dynamic Groups Another feature of our application is that we expect the system (at least in the first years of installment) to often add new group members, while we consider revocations of credentials to be less frequently required. XSGS accounts for this as joins of new group members (unlike revocations) do not require updates of the group public key. We stress that if member joins do not require modifications of GPK, it is impossible to not modify the group public key when revoking users.

Revocation in XSGS Revoking group members (i.e., their credentials) requires special care in group signatures. This is because a group member does never reveal its certificate in the clear. The approach underlying XSGS is very efficient. It is based on dynamic accumulators [37, 38]. Briefly speaking, first the group manager modifies GPK such that all credentials become invalid. Next it publishes a set of value that helps every user except for the one that should be revoked to update their credentials, so that they become valid for the new GPK again. The user to be revoked cannot update its credential in this way and is in the following not able to produce the NIZK part of the group signature anymore. The revocation mechanism in XSGS has two major benefits. First, for each revoked user there is only a single, small, and constant set of values that has to be transferred to the group members to enable them to update their credentials. Second, this information does not have to be transferred in secret. Instead the group manager it can simply be made publicly available by the group manager.

3 System Design

In this section we describe all processes that constitute our system.

3.1 Bootstrapping the System

Before we can start authenticating users, charging vehicles, and securely transmitting energy consumption data, we have to set up the infrastructure. The clearing house serves as trust anchor (RootCA) within the RSA-based PKI used for user authentication. Each electric utility that

cooperates in this system serves as Intermediary CA and can thus issue certificates to its customers. We assume that each customer holds a secret signing key sk together with its corresponding public verification key pk, a certificate chain $cert_{CUS}$, and the public verification key of the RootCA PK_{CA}. Similarly we suppose that each CS holds a secret key USK, public key UPK, a certificate chain $Cert$, and PK_{CA}. Customers can use their certificate chains to authenticate themselves towards any CS in the system. Figure 2 illustrates the architecture.

The clearing house also acts as the *group manager* within the XSGS scheme. It can add a new CS to the group by issuing a certificate (credential) $UCert$ to CS. A CS with a valid $UCert$ is also referred to as *group member*. The clearing house can also revoke the ability of group members to sign on behalf of the group. An entity sufficiently independent of the clearing house serves as the *opener*. As sketched above, this could be a corporate counsel at the clearing house or an independent notary, i.e., an entity that can be trusted to act lawfully. In our scenario N electric utilities choose to cooperate by utilizing a certain clearing house. Each utility i provides m_i charging stations to the public (cf. Figure 1).

In order to bootstrap the XSGS scheme, the group manager first needs to generate the group (curve) parameters of a bilinear group (including group descriptions, generators, and pairing specification). Technically, the bilinear group consists of two elliptic curve groups $\mathbb{G}_1$ and $\mathbb{G}_2$ of prime order p with random generators $G_1 \in \mathbb{G}_1$ and $H, G_2 \in \mathbb{G}_2$ and the description of a non-degenerated bilinear pairing $e : G_1 \times G_2 \to G_t$ such that $e(G_1^a, G_2^b) = e(G_1, G_2)^{ab}$ for every $a, b \in \mathbb{Z}_p$. For more details we refer to the paper by Boneh et al. [32], however, we illustrate the process in Figure 3.

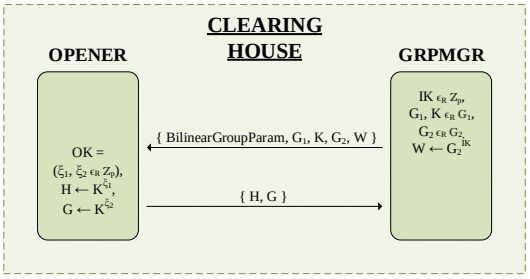

Figure 3: Bootstrapping of XSGS

Next it generates a secret Diffie-Hellman key $IK \in \mathbb{Z}_p$ (called issuer key) with its corresponding public key $W = G_2^{IK}$. The issuer key IK is used to generate certificates for new group members. Given these values, the opener generates a private key of a chosen-ciphertext secure encryption system, the opening key OK. The corresponding public encryption key is denoted as OPK. The public key OPK is used in the signing process of the group signature scheme to encrypt the signer's certificate $UCert$. This enables the opener to reveal which CS has actually created a given group signature. On a technical level OK consist of two independent secret keys of an ELGamal encryption system. OPK contains the corresponding public keys. It is well known that ElGamal is only chosen-plaintext secure. However, the system applies the well-known Naor-Yung transformation [39] which encrypts a given messsage under both ElGamal keys resulting in ciphertext Z_1 and Z_2. Additionally, it generates a NIZK proof P of equality of plaintexts in Z_1 and Z_2. The ciphertext Z consist of $Z = (Z_1, Z_2, P)$. The group public key GPK consist of the paramaters of the bilinear group, W, and OPK.

3.2 Setting Up New Charging Stations

Each new CS must join the group before it can sign metering data. Now that group manager and opener are set up, the group manager can add new charging stations to the group. Note that all charging stations, independent of the utility that operates them, will be members of the same group. We illustrate the algorithmic details of the join process in Figure 4.

The group manager starts the join process by transmitting the GPK to the CS. The CS draws its private signing key $UK \in \mathbb{Z}_p$ and computes a commitment $C = H^{UK}$ of UK. Then it sends C together with a NIZK proof of knowledge of UK to the group manager. On successful verification of this proof, the group manager selects a random signing key $x \in \mathbb{Z}_p$ for the CS and calculates the group member identifier

$$A = (G_1 \cdot C)^{\frac{1}{IK+x}} \Leftrightarrow e(A, W \cdot G_2^x) = e(G_1 \cdot C, G_2).$$

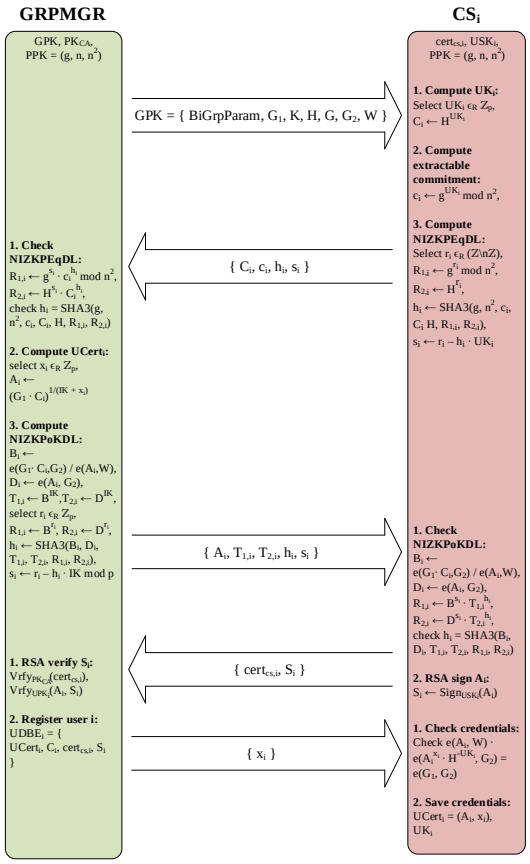

Figure 4: XSGS Join Procedure illustrated

The values A and x constitute the certificate $UCert$ of the CS. Intuitively, $UCert$ is a digital signature over x that can only be computed with the help of IK. The group manager first sends A to the CS and proofs that it knows a corresponding x that fulfills the above equation. Knowing that its communication partner can indeed issue certificates, the CS produces a classical signature S using its USK over A as $S = Sign_{USK}(A)$ and sends $(S, cert_{CS})$ to the issuer. This pair is important when resolving disputes as it binds the anonymous certificate $UCert$ to a concrete CS that can be identified via the classical PKI. If the signature is valid, the group manager sends x to the CS and registers the entry $(UCert, C, cert_{CS}, S)$ in a database.

Now since $C = H^{UK}$ and UK is known to the CS we get that

$$A = (G_1 \cdot H^{UK})^{\frac{1}{IK+x}}$$
$$\Leftrightarrow$$
$$e(A, W \cdot G_2^x) = e(G_1 \cdot H^{UK}, G_2).$$

3.3 Decommission of Charging Stations

Occasionally it may be necessary to remove a CS from the group, be it because it is replaced by a CS of a newer generation or to deal with a compromise. We consider the revocation of a group member's credentials to be a less frequent event than the joining of a new member. Thus, while $UCert$ and UK remain unchanged upon the joining of a new member, removing a member from the group requires that all remaining group members receive information on how to re-calculate their group identifiers A.

The process is illustrated in Figure 5. Assume the group manager wants to revoke a CS with $UCert' = (A', x')$. First, it publishes an updated version of the GPK. For example G_1, G_2, and H are substituted by $G_1^* = G_1^{\frac{1}{IK+x'}}$, $G_2^* = G_2^{\frac{1}{IK+x'}}$, and $H^* = H^{\frac{1}{IK+x'}}$. Next each group member with $UCert = (A, x)$ and secret key UK except for the one to be revoked has to

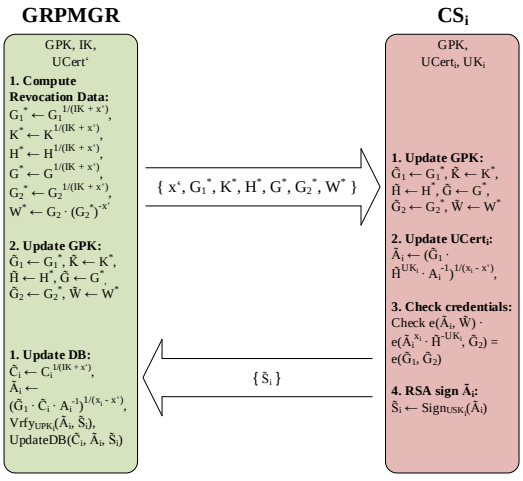

Figure 5: XSGS Revoke Procedure illustrated

update its group identifier A to $A^* = A^{\frac{1}{IK+x'}}$. To this end it is sufficient that the group manager simply publishes x'.

$$A^* = A^{\frac{1}{IK+x'}} = \left(G_1^* \cdot H^{*UK} \cdot A^{-1}\right)^{\frac{1}{(x-x')}} .$$

Next, each charging station computes a new signature $S^* = Sign_{USK}(A^*)$ over the new group member identifier A^* and sends it to the group manager. The group manager verifies S^* from each CS and, on success, updates the existing database entries with the new values for A^*, C^* and S^*. Note that the CSs do not have to save an incremental revocation list of all revoked members to decide on the validity of newly signed metering data. However, it might be necessary for the group manager to retain a limited set of old group credentials for the time span that the respective jurisdiction sets for the resolution of disputes concerning past charging processes.

3.4 User Authentication

Before the CS channels electricity into the EV, the (owner of the) EV has to authenticate itself to the CS for proper billing. At the same time the CS should authenticate itself towards the (owner of the) EV to prove that it is genuine. This may thwart attacks where a rogue CS may be setup that exploits access to an EV in a malicious way.

Due to its severely limited resources we cannot use the JCOP card to efficiently establish a mutually authenticated TLS session with the CS. Instead we have to devise a more cost-efficient design. Luckily, in the authentication process only few additional data except for the authentication information needs to transferred from the CS to the EV and vice versa.

To authenticate each other, a smartcard (equipped with secret key sk, public key pk, and certificate chain $cert_{CUS}$) and a CS can engage in the following protocol:

1. The CS draws a uniformly random bitstring r_{CS} and sends it over to the EV.
2. The customer's smartcard responds with a uniformly random bitstring r_{CUS}.
3. Using USK, the CS computes a signature over the concatenation of r_{CS} and r_{CUS}: $s_{CS} = Sign(USK, r_{CS}||r_{CUS})$. Next, s_{CS} is sent together with $cert_{CS}$ to the smartcard.
4. The smartcard first verifies $cert_{CS}$ with respect to PK_{CA}. On success it verifies s_{CS} using UPK. If this check is successful as well, the smartcard computes $s_{CUS} = Sign(sk, r_{CS}||r_{CUS}||s_{CS})$ and sends s_{CUS} together with $cert_{CUS}$ to the CS. If the signature verifications fails, the smartcard aborts.
5. The CS checks whether $cert_{CUS}$ and s_{CUS} are valid signatures with respect to PK_{CA} and pk respectively. On failure it aborts.

The above protocol is a classical challenge and response protocol. Intuitively, it exploits that only the holder of the secret signing key can produce valid signatures for arbitrary messages. The intial exchange of nonces thwarts replay attacks.

Instead of equipping customers with smartcards, the smartcard can also be bound to and incorporated in the EV to allow for ISO 15118 compliant authentication [40], where the vehicle (and not the customer) authenticates towards the CS. Communication then takes places using a specially equipped power cable instead of a contactless interface.

3.5 Ensuring Authenticity of Metering Data

When the charging process is terminated (i.e., the cable connection between EV and CS is severed), the CS creates a message M consisting of the authenticated customer identity (as derived from the authentication process described in Section 3.4), the amount of energy consumed by the customer, two timestamps marking the beginning and the end of the charging process, and a string that identifies the utility owning the CS. In some legislations, standards and measurements laws may require the transmission and storage of the identifier ($meterID$) of the calibrated meter used

to determine the energy consumption or similar information. However, including this $meterID$ in the message would reveal the user's location, as any electric utility can be expected to know the physical location of each of its meters. To avoid this, we have to adapt the group signature scheme slightly. Instead of being sent in the clear, the $meterID$ is encrypted using the opener's encryption key OPK before being added to M. In the same way other location-critical information can be incorporated into the group signature. Only the opener can decrypt these values using its secret decryption key OSK. We stress that while the $meterID$ is always encrypted with the opener's public key and never transmitted in the clear, it is not neccessary to prove that the *correct* $meterID$ has been incorporated into the ciphertext. The opener can uniquely identify the CS and any incorrect information of a CS on its $meterID$ can thus easily be revealed. As sketched above, CS's group signature s on M consists of an encryption Z of $UCert$ and a message-dependent NIZK proof showing that CS knows a valid $UCert$ with corresponding UK which fullfil Equation 3.2 and that $UCert$ has been encrypted correctly under public key OPK in the ciphertext Z (which is part of s). Intuitively, these types of message-dependent proofs work like signatures. Generating them on new messages requires the creator to know A, x and UK. They are often referred to as signatures of knowledge [41]. XSGS uses particularly efficient NIZKs that can be computed by applying the well-known Fiat-Shamir heuristic [42] to interactive zero-knowledge protocols in the random oracle model [43]. On a more technical level the proof shows that its creator knows several discrete logarithms; x, UK, and the secret exponents used to encrypt A with the ElGamal encryption systems. Intuitively, it proves that if the ElGamal ciphertexts were decrypted, then the resulting plaintext would, together with x, UK fulfill Equation 3.2. For more details on the computations of the group signature, we refer to the literature [31, 34].

3.6 Transmission of Metering Data

In the next step, the CS needs to transmit the message (M, s) to the clearing house. To ensure that the user can be billed correctly, the CS needs to guarantee that each message reaches the backend (BE). For this we rely on TLS. We use a ciphersuite based on Ephemeral Diffie-Hellman (DHE) with CBC-MAC, as it offers perfect forward secrecy and because of its cryptographic security properties: it has recently be shown to be provably secure in a strong security model [44]. Besides ensuring the confidentiality of the transmitted data, the application of TLS allows for the CS to determine that the BE did indeed receive the message. This is crucial, as without the BE receiving (M, s), the user will not be billed and the utility operating the CS will lose revenue.

To prevent the disclosure of the CS's network location, the CS first connects to the Tor network and establishes a routing circuit. It then starts a TLS session with the BE and in the process verifies the certificate presented by BE. Upon successful establishment of the TLS tunnel through the Tor circuit, the CS transmits (M, s). The BE acknowledges the successful submission by sending the string `ACK` and a timestamp. We rely on TLS for the authenticity of the reply.

3.7 Verification of Metering Data

When the BE at the clearing house has received (M, s) it verifies the group signature s by checking the NIZK proof with respect to the GPK and thus determines whether the consumption data that is bound to the identity of a customer is valid. For details on the computations, we refer to [31,34]. If the signature does not verify it simply discards the message as it cannot stem from a CS within the group. On success, the signed tuple M is passed on to the clearing service for processing. As there is one central verifier in the system that verifies all metering data, batch verification of group signatures offers a significant efficiency gain.

3.8 Dispute Resolution

In the case of a dispute, the opener can craft a non-repudiable publicly verifiable proof of the actual creator of a given group signature. The opener will act so only on the request of a judge or with the consent of the customer. Please note that even after a message M_i has been subject to

the opening process, it is impossible to decide, whether a CS, who signed M_i also signed a different message M_j, i.e, the location of other, potentially unrelated charging events remains hidden. To open the signature s, the opener uses its secret opening key OK to decrypt the ciphertext Z and obtain the certificate $UCert$ of the signer. Next it uses its access to the registration database to obtain UPK and S which correspond to $UCert$. From this information she computes a publicly verifiable NIZK proof that $UCert$ is actually encrypted in Z. Together with the database entry $A, cert_{CS}, S$ this convincingly reveals the identity of the signer in a non-repudiable way.

4 Evaluation

We now describe how we evaluated our prototype implementation. Furthermore, we also present an overview of the performance results obtained both for the various operations of the XSGS scheme and the transmission of data from a CS to the BE.

4.1 Evaluation Environment

We aim at evaluating our approach in an realistic environment. Thus, we implemented XSGS completely and tested the creation of signed messages, the setup process for adding new charging stations and the procedure to decommission charging stations on a prototype of a CS for EVs built at our department. The CS contains an inexpensive industrial-grade Intel Atom platform (CS1, cf. Table 1) as control unit that interacts with the energy flow control subsystems within the CS and acts as a front-end to the user. Additionally, we evaluated our implementation on a Freescale i.MX53, which is an implementation of an ARM A8 core. Comparable platforms to both variants can be found in CSes in the market or under development today.

As BE we chose an Intel server platform (cf. Table 1). We used this platform to evaluate all XSGS operations typically performed by the group manager, opener, judge, or any entity that wishes to verify a signature. We also created signatures and performed join operations as a comparison to the measurements on the actual CS. While the Tor network is widely used and considered usable for non-time critical applications, we also used this platform to evaluate if latency and throughput are acceptable in our application scenario.

	Hardware Platform	OS
CS_1	Intel Atom D2550, 1GB RAM	Ubuntu 12.04
CS_2	Freescale i.MX53, 1GB RAM	Ubuntu 10.04
BE	Intel Xeon X5650, 2GB RAM	Ubuntu 12.04

Table 1: Evaluation Environment

4.2 Evaluation Results

While in some scenarios it might not be necessary for the join operation to be performed between CS and BE during the setup procedure (but rather between BE and the entity that supplies the key material to the CS, e.g., on a smartcard), from a performance point of view this is entirely feasible. We performed the setup procedure required for adding a new CS 100 times. The computations necessary on the CS are performed on average in 757.43 ms on CS_1 and 1077.29 ms on CS_2, while the computations on the BE took 55 ms on average. Accordingly, we performed 100 decommission procedures: on average, the computations performed on CS_1 take 48.99 ms (resp. 77.78 ms on CS_2), while the computations performed on the BE take 20 ms. We also performed 100 dispute resolution procedures on the BE: on average opening a message takes 8.2 ms, while judging takes 6.9 ms.

We evaluate the time required to prepare a message to transmit the metering data to the BE. Preparing a message containing 1000 bytes (taken from `/dev/urandom`) takes 28.50 ms on average

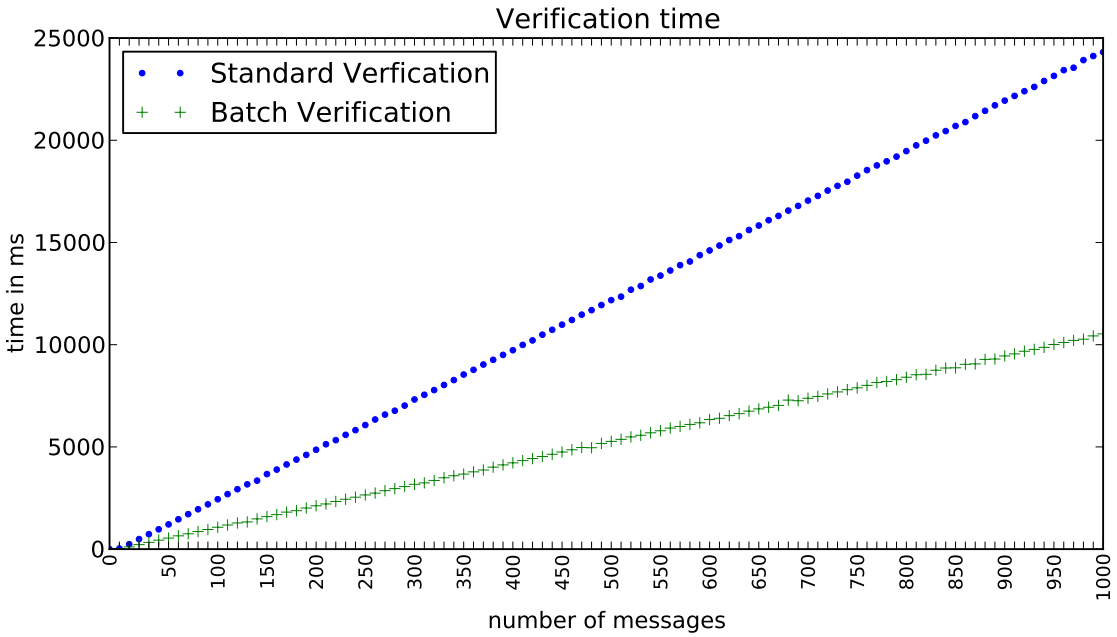

Figure 6: Time required for verification by #messages received

on CS_1; on CS_2 the process takes 41.50 ms. Preparing a message that allows for batch verification on the BE takes slightly longer: we require 28.79 ms on CS_1 and 43.11 ms on CS_2. In both cases the time required for preparing the messagess scales linearly with the amount of messages. We also evaluated whether an increased message size significantly increases the time to create a signed message on limited hardware. We created 100 messages of each size. Figure 7 shows that the size of the message only has a limited impact on the time required to create a valid signature. This is expected, as we sign the SHA-3 hash of the message. For a message size up to 100,000 bytes message creation takes less than 33 ms on CS_1 and 54.17 ms on CS_2, and increases only slightly with message size. Creating a signed message of one million bytes takes 66.68 ms on average on CS_1 and 161.12 ms on CS_2. These results show that ensuring the authenticity of messages by means of group signatures is feasible on the limited hardware found in a CS. Even more so, as we only need to generated one signature for each charging process. As even quick charging takes about 30 minutes for a full charge today and will take at least minutes in the foreseeable future, the amount time required for signing the customer's energy consumption data is insignificantly small. As transmission times vary due to network latency, we evaluate the network performance separately.

Creating a batch verification-enabled message only requires very little more time than creating a normal message. However, being able to batch verify messages offers a significant performance increase. While a CS will typically only creates one message every few minutes or every few hours, each message has to be verified by the BE. The verification of a normal message takes 30 ms, a single batch-enabled messages can be verified in about the same time. Figure 6 shows that the time required for verification increases linearly with the amount of messages. Standard verification allows for processing 41 messages per second on the BE, while batch verification allows for processing of 93 messages in the same time. When comparing the time required for verifying one thousand messages, batch verification is about 2.3 times faster. In a worst case scenario, where a batch contains so many invalid signatures that it is faster to verify each individual message, we can still process more than 148,000 messages per hour using a single CPU core. As the process can be parallelized at will, a comparable server with eight CPUs cores instead of one is sufficient for processing more than one million messages per hour.

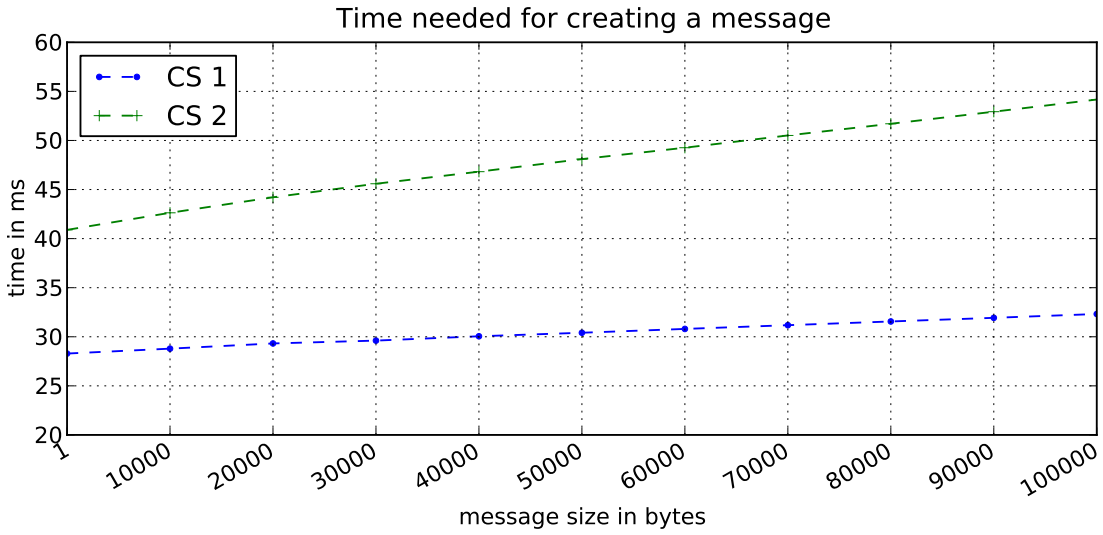

Figure 7: Time required for message creation by size

We used `iperf`[3] to measure whether the Tor network offers enough bandwidth for transmitting metering data from the CS to the BE. We controlled that the bandwidth between the host running the `iperf` server and and the one running the client is not the limiting factor and repeated our measurements at various times of the day, building a new Tor circuit for each iteration. We were able to transfer a minimum of 373 kbit per second and a maximum of 1.07 Mbits per second through the Tor network. While the actual throughput may vary depending on the time of day and the chosen circuit, our evaluation shows that it is reasonable to assume that we can transfer metering data through the Tor network, especially as the communication between CS and BE is not subject to real-time requirements.

In summary, we found that our approach performed well on all tested platforms and, most importantly, is fast enough for our application.

5 Discussion

In this section we discuss possible attacks against both the authenticity of billing-relevant data and against the user's location privacy, and discuss an alternative application scenario.

5.1 Malicious Customer

While our system is well equipped to counter attackers with capabilities as described in Section 2.1, there exists the theoretical possibility that an attacker, who is a valid customer in the system, could force a CS_1 offline before a revocation of a different CS_2 takes place. Thus CS_1 does not realize that the group credentials have changed and must be recomputed. The attacker then authenticates herself and charges her EV at CS_1, which is possible as user authentication works offline. The CS signs the metering data with its current credentials. At some point in the future, when the CS is online again, it transmits the data to the BE. It will then also receive new group credentials and will be able to create valid messages, as during the revocation process. Still, the BE will discard the delayed metering data from the CS as it has been generated with the old credentials. Hence, the attacker was able to charge for free in the meantime. There are at least two counters to this attack. First, if the CS is up and running again, it may simply re-sign all the unsent metering data with the updated credential. Second, if the CS is for some reason not able

[3] http://iperf.sourceforge.net/

to continue signature generation (e.g., a trusted key storage is broken), we can still use the old group signature to bill the customer correctly.

To be able to resolve disputes that concern metering data that was signed at some point in the past before one or more revocation events took place, we have to store old group credentials for the period regulations or utility terms and conditions dictate for dispute resolution. Thus, as old credentials are retained anyway, we can verify incoming metering data with the credentials that were valid at the time of signature creation. For operational reasons, most utilities will want to define a maximum time a CS may be offline before it is marked as faulty in a monitoring system and a repair crew is sent out. While the system is designed to be tolerant towards network outages, long periods where a CS is offline will be undesirable for management reasons. As a consequence, there will be an archived GPK available to verify the incoming message, as the dispute resolution period is longer than the maximum offline time tolerated by the utility operating the CS.

5.2 Tracking and Localization Attacks

Ma et al. showed that if a set of traces of time and corresponding location of mobiles nodes exist, where "[t]he traces are anonymous in that the true identity of a participant has been replaced by a random and unique identifier" [45], a small amount of side information is sufficient for an attacker to infer the true identity of a user. The work of de Montjoye et al. [46] supports these claims and shows that even datasets with coarse traces provide little anonymity: the authors demonstrate that four spatio-temporal points are enough to uniquely identify 95% of the individuals.

However, none of these attacks is applicable to our system. We do not aim at protecting the *identity* of the user, but we aim at protecting the user's *location*. All information is transmitted encrypted with a provably secure TLS variant. Thus the attacker needs to be a legitimate receiver of the data, i.e., the clearing house or a utility. Both receive the following information: customer A of utility B consumed N kWh of energy, starting from timestamp X, ending at timestamp Y. Every location-bound token, like the CS's public key and the meterID, is encrypted only to the opener and thus never leaked to any other party. This encrypted data is also transmitted to both the clearing house and the respective utility. However, it is meaningful to neither party as both lack the appropriate key to decrypt the data.

The data available to an adversary thus does not contain the location of the user, nor can the attacker use the amount of energy consumed to infer the distance the user has driven between two charging events. For instance, wind resistance increases with the speed of a vehicle, such that a user can cover a long distance at lower speed or a shorter distance at higher speed while consuming the same amount of energy. An attacker may infer a limited amount of information from the timestamps written at the beginning and the end of the charging process, namely how often a user charged her vehicle and how long the individual charging processes took. However, it is indiscernible to the attacker whether these charging processes took place at different CSs or always at the same CS. The attacker also still lacks the information of where the relevant CSes are located (assuming there is more than one CS within reachable distance of the user).

Shokri et al. [47] propose a metric to quantify the performance of a location privacy protection mechanism (LPPM) that, given a trace of spatio-temporal locations, protects the user from localization attacks, meeting disclosure attacks and aggregated presence attacks by reducing the accuracy and/or precision of the events' spatio-temporal information. Our systems applies location hiding as an online LPPM in a distributed architecture, i.e., we only look at the current event at the time of its creation and hide all location-bound information by encrypting it to the opener. As argued above, while records of user interaction exists for billing purposes, they do not contain any spatio-temporal locations or references to such data. This means that an adversary who knows the location of every CS, may determine the location where the EV *could have been charged* with a high accuracy (as it was necessarily at the location of a CS), but she is unable to achieve a high correctness as to where the EV *was actually charged*.

A potential information leak could exist if the billing data that the clearing house receives from a charging station not only contains the information that a user is a customer of a given utility, but also contains the information which utility owns the CS this customer just used. For

example, given two charge events at two different utilities separated by two hours, there might be only one possible pair of charging stations for which this would have been feasible. However, the clearing house does not receive this information. It can merely receive the information that a user is a registered customer of a given utility. At the end of the clearing period the clearinghouse also receives the accumulated amount of energy each utility dispensed via its CSs and can thus balance claims against each other.

5.3 Alternative Application Scenario

While we aim at protecting location privacy in a multi-utility setup where the customer can roam between energy providers, our system can also help to protect the location privacy of customers of a single, isolated energy provider. In the short term, this scenario is not far-fetched, as a utility that, for example, can offer a unique benefit to its customers, like a well laid-out network of quick charging CSes, may currently not feel the need to offer (inbound) roaming.

If, for instance, this company wants to guarantee that only one entity within the company can disclose the movement profile of a customer, instead of a multitude of employees, our system can easily be adapted to this setup. The benefit for both the customer and the company is that the latter can now provide a credible privacy policy. The party within the company that acts as the opener can be clearly defined and thus has credible, technical means to enforce the privacy policy. However, as with our original scenario, the opener must still be trusted to act in compliance with local regulation and, most importantly in this alternative scenario, the privacy policy as agreed upon between utility and customers.

6 Related Work

Location privacy has been recognized as being desirable as early as 1996, when Jackson [48] proposed a modification to the Active Badge in-building localization system, to enable a user to control who can access her location information. In the field of pervasive computing, the importance of location privacy has also been recognized, for example, by Beresford and Stajano [49] and also in the context of location-based mobile applications [50]. The importance of location privacy in the context of transportation is underlined by numerous publications that aim at preserving location privacy in various applications like vehicular communication systems [51–55], ticketing for public transport systems [56–58], and electronic road toll collection [59–61]. In this context, Chen et al. [62] proposed to use a group signature scheme to protect the privacy of users of an electronic toll pricing system. However, Chen et al. choose to remain at an abstract level, while we adapt, implement, and evaluate a carefully-chosen group signature scheme to our application.

While all of the above publications target different fields of application, a limited amount of publications have considered location privacy in the context of e-mobility so far: Chao Li [63] explored the efficiency of the Compact e-Cash scheme by Camenisch et al. [64] on ARM devices by applying it to a payment scheme for EV charging stations. Liu et al. [65] propose a anonymous electronic payment scheme that supports two-way anonymous payments. The authors employ the BBS+ signature scheme [66] that is derived from the BBS group signature scheme [32] to implement *revocable* anonymity to prevent cheating and double-spending, as well as, a judging authority to resolve disputes. The approach is similar to our work as it incorporates a judging entity for dispute resolution and the possibility to revoke anonymity, if necessary. However, while Liu et al. aim at protecting the user's location privacy by not disclosing her identity, our approach protect the user's location privacy by not disclosing her location. This allows our approach to be used in applications where the anonymous usage is not an option for the infrastructure provider. In contrast to Li, as well as, Liu et al. we do not aim at providing a payment solution, but aim at providing authentic energy consumption data, cryptographically bound to the user's identity, as basis for an arbitrary billing process, while at the same time protecting the customer's location privacy.

Stegelmann and Kesdogan [67] approach the topic of smart grid privacy, actively researched in the field of smart metering (e.g., by Cavoukian et al. [68]), from the mobile perspective of EV users. From a smart grid perspective, EV batteries can be seen as an energy buffer that can help stabilize the grid [69]. Stegelmann and Kesdogan aim at enabling a smart grid to manage these location-variable ressources while preventing the unnecessary disclosure of location information. To prevent the disclosure of the user's location, the authors propose the use of a anonymity network, like Tor [27]. To provide accountability, integrity and non-repudiation of messages the authors propose to use a anonymous credential system, like idemix [70], which also builds upon ideas found in the construction of group signature schemes [71]. Stegelmann's and Kesdogan's approach aims at providing location privacy in the presence of a smart grid. However, electric vehicles and charging infrastructure exist today, while it is not yet clear when a wide-spread smart grid infrastructure is bound to exist. Also, the amount of charging cycles an EV battery can endure without degenerating in capacity are limited. At the time of writing, batteries make up about a third of the net worth of an EV [72]. This is unlikely to change until there is a completely different battery technology. Customers can not be expected to accept technology that decreases the lifetime of their single most valuable investment in an EV. Thus, our approach aims to preserve the user's location privacy *now*, within the boundaries of currently deployed technology, while being open to adaption to future infrastructure. In contrast to both Liu et al. (who only assess time required for exponentiation and pairings) and Stegelmann and Kesdogan, we implement our system and evaluate it under realistic conditions.

7 Conclusion

In this paper, we introduced a system that enables location privacy for the whole charging process of electric vehicles. Such an approach is necessary to address the problem of location privacy in the upcoming world of electric vehicles, when anonymous usage of charging stations is not an option to the vendor. Our system also fully supports all requirements needed to bill the customer after the charging process and enable users to roam between different charging stations provided by different electric utilities. As such, our system covers all relevant aspects required for the charging of electric vehicles. The basic idea of our approach is to adapt a group key signature scheme to the settings of electric vehicles. We described all protocol steps and outlined how the system can be deployed in practice. In an empirical evaluation, we also demonstrated that the scheme has a low overhead and can scale to millions of charging processes per hour (even on off-the-shelf hardware).

Acknowledgments

This work was supported by the German Federal Ministry of Economics and Technology (Grant 01ME12025 SecMobil).

References

[1] Gijs Mom, *The Electric Vehicle: Technology and Expectations in the Automobile Age.* John Hopkins University Press, 2004.

[2] HybridCars.com, "December 2012 dashboard," http://www.hybridcars.com/december-2012-dashboard, 2012. [Online]. Available: http://www.hybridcars.com/december-2012-dashboard

[3] Pike Research, "Electric vehicle market forecasts," http://www.pikeresearch.com/research/electric-vehicle-market-forecasts, 2013. [Online]. Available: http://www.pikeresearch.com/research/electric-vehicle-market-forecasts

[4] PlugShare, "FAQ," http://www.plugshare.com/tpl/faq_popup.html, 2013. [Online]. Available: http://www.plugshare.com/tpl/faq_popup.html

[5] LEMnet, "Map of charging stations," http://www.lemnet.org/LEMnet_Map.asp, 2013. [Online]. Available: http://www.lemnet.org/LEMnet_Map.asp

[6] cars21.com, "EU proposes minimum of 8 million EV charging points by 2020," http://beta.cars21.com/news/view/5171, 2013.

[7] Tesla Motors, "Supercharger," http://www.teslamotors.com/supercharger, 2013. [Online]. Available: http://www.teslamotors.com/supercharger

[8] e-laad.nl, "Over stichting e-laad," http://www.e-laad.nl/onze-partners/2-deelnemers-aan-e-laad, 2013.

[9] Charge Point America, "Electric vehicle charging by ChargePoint," http://www.chargepoint.net/, 2013. [Online]. Available: http://www.chargepoint.net/

[10] Blink, "Commercial chargers," http://www.blinknetwork.com/chargers-commercial.html, 2013.

[11] SemaConnect, "SemaCharge," https://semacharge.com/, 2013.

[12] Andrew J. Blumberg and Peter Eckersley, "On locational privacy, and how to avoid losing it forever," Electronic Frontier Foundation, Tech. Rep., 2009. [Online]. Available: https://www.eff.org/wp/locational-privacy

[13] C. Paar, K. Schramm, A. Weimerskirch, and W. Burleson, "Securing green cars: IT security in next-generation electric vehicle systems," in *Annual Meeting and Exposition of the Intelligent Transportation Society of America*, 2009.

[14] Christina Hager, "Divorce lawyers using fast lane to track cheaters," http://msl1.mit.edu/furdlog/docs/2007-08-10_wbz_fastlane_tracking.pdf, 2007.

[15] P. F. Riley, "The tolls of privacy: An underestimated roadblock for electronic toll collection usage," *Computer Law & Security Review*, vol. 24, no. 6, pp. 521–528, 2008. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0267364908001349

[16] Pacific Gas and Electric (PG&E), "Rate options," http://www.pge.com/en/about/rates/rateinfo/rateoptions/index.page, 2013.

[17] OCPP Forum, "OCPP 1.5 interface description between charge point and central system FINAL," http://www.ocppforum.net/sites/default/files/ocpp_specification_1.5_final.pdf, 2012.

[18] R. L. Rivest, A. Shamir, and L. Adleman, "A method for obtaining digital signatures and public-key cryptosystems," *Commun. ACM*, vol. 21, no. 2, Feb. 1978. [Online]. Available: http://doi.acm.org/10.1145/359340.359342

[19] BSI, "Certification report BSI-DSZ-CC-0426-2007," https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/Zertifizierung/Reporte04/0426a_pdf.pdf?__blob=publicationFile, Tech. Rep., 2007.

[20] D. Chaum, A. Fiat, and M. Naor, "Untraceable electronic cash," in *Advances in Cryptology - CRYPTO*, 1988.

[21] D. Chaum, "Security without identification: transaction systems to make big brother obsolete," *Commun. ACM*, vol. 28, no. 10, p. 10301044, Oct. 1985. [Online]. Available: http://doi.acm.org/10.1145/4372.4373

[22] S. Brands, "Electronic cash systems based on the representation problem in groups of prime order," in *CRYPTO*, 1993.

[23] J. L. Camenisch, J.-M. Piveteau, and M. A. Stadler, "An efficient electronic payment system protecting privacy," in *ESORICS*, 1994.

[24] S. von Solms and D. Naccache, "On blind signatures and perfect crimes," *Computers & Security*, vol. 11, no. 6, pp. 581–583, Oct. 1992. [Online]. Available: http://www.sciencedirect.com/science/article/pii/016740489290193U

[25] P. C. Johnson, A. Kapadia, P. P. Tsang, and S. W. Smith, "Nymble: Anonymous IP-Address blocking," in *Privacy Enhancing Technologies*, 2007.

[26] P. P. Tsang, M. H. Au, A. Kapadia, and S. W. Smith, "BLAC: revoking repeatedly misbehaving anonymous users without relying on TTPs," *ACM Trans. Inf. Syst. Secur.*, 2010.

[27] R. Dingledine, N. Mathewson, and P. Syverson, "Tor: the second-generation onion router," in *13th USENIX Security Symposium*, 2004.

[28] D. Chaum and E. van Heyst, "Group signatures," in *EUROCRYPT*, 1991, pp. 257–265.

[29] S. Goldwasser, S. Micali, and C. Rackoff, "The knowledge complexity of interactive proof systems," *SIAM J. Comput.*, vol. 18, no. 1, pp. 186–208, 1989.

[30] M. Blum, A. D. Santis, S. Micali, and G. Persiano, "Noninteractive zero-knowledge," *SIAM J. Comput.*, vol. 20, no. 6, pp. 1084–1118, 1991.

[31] C. Delerablée and D. Pointcheval, "Dynamic fully anonymous short group signatures," in *VIETCRYPT*, 2006, pp. 193–210.

[32] D. Boneh, X. Boyen, and H. Shacham, "Short group signatures," in *CRYPTO*, 2004, pp. 41–55.

[33] M. Bellare, H. Shi, and C. Zhang, "Foundations of group signatures: The case of dynamic groups," in *CT-RSA*, 2005, pp. 136–153.

[34] K. Kim, I. Yie, S. Lim, and D. Nyang, "Batch verification and finding invalid signatures in a group signature scheme," *I. J. Network Security*, vol. 13, no. 2, pp. 61–70, 2011.

[35] A. Fiat, "Batch rsa," in *CRYPTO*, ser. Lecture Notes in Computer Science, G. Brassard, Ed., vol. 435. Springer, 1989, pp. 175–185.

[36] M. Bellare, J. A. Garay, and T. Rabin, "Fast batch verification for modular exponentiation and digital signatures," in *EUROCRYPT*, 1998, pp. 236–250.

[37] J. Camenisch and A. Lysyanskaya, "Dynamic accumulators and application to efficient revocation of anonymous credentials," in *CRYPTO*, 2002, pp. 61–76.

[38] L. Nguyen, "Accumulators from bilinear pairings and applications," in *CT-RSA*, 2005, pp. 275–292.

[39] M. Naor and M. Yung, "Universal one-way hash functions and their cryptographic applications," in *STOC*, 1989, pp. 33–43.

[40] International Organization for Standardization, "ISO/IEC DIS 15118 - road vehicles – vehicle to grid communication interface," 2012. [Online]. Available: http://www.iso.org/iso/catalogue_detail.htm?csnumber=55366

[41] M. Chase and A. Lysyanskaya, "On signatures of knowledge," in *CRYPTO*, 2006, pp. 78–96.

[42] A. Fiat and A. Shamir, "How to prove yourself: Practical solutions to identification and signature problems," in *CRYPTO*, 1986, pp. 186–194.

[43] M. Bellare and P. Rogaway, "Random oracles are practical: A paradigm for designing efficient protocols," in *ACM CCS*, 1993.

[44] T. Jager, F. Kohlar, S. Schäge, and J. Schwenk, "On the security of TLS-DHE in the standard model," in *Advances in Cryptology - CRYPTO*, 2012.

[45] C. Y. Ma, D. K. Yau, N. K. Yip, and N. S. Rao, "Privacy vulnerability of published anonymous mobility traces," in *MobiCom '10*, 2010.

[46] Y.-A. de Montjoye, C. A. Hidalgo, M. Verleysen, and V. D. Blondel, "Unique in the crowd: The privacy bounds of human mobility," *Scientific Reports*, 2013. [Online]. Available: http://www.nature.com/srep/2013/130325/srep01376/full/srep01376.html

[47] R. Shokri, G. Theodorakopoulos, J. Le Boudec, and J. Hubaux, "Quantifying location privacy," in *2011 IEEE Symposium on Security and Privacy (SP)*, May 2011.

[48] I. Jackson, "Anonymous addresses and confidentiality of location," in *Information Hiding*, 1996.

[49] A.R. Beresford and F. Stajano, "Location privacy in pervasive computing," *IEEE Pervasive Computing*, vol. 2, no. 1, pp. 46 – 55, Mar. 2003.

[50] R. A. Popa, A. J. Blumberg, H. Balakrishnan, and F. H. Li, "Privacy and accountability for location-based aggregate statistics," in *ACM CCS*, 2011.

[51] J.-P. Hubaux, S. Capkun, and J. Luo, "The security and privacy of smart vehicles," *Security & Privacy, IEEE*, vol. 2, no. 3, pp. 49–55, 2004.

[52] F. Dötzer, "Privacy issues in vehicular ad hoc networks," in *Privacy Enhancing Technologies*, 2006.

[53] J. Freudiger, M. Raya, M. Félegyházi, P. Papadimitratos *et al.*, "Mix-zones for location privacy in vehicular networks," in *Win-ITS*, 2007.

[54] K. Sampigethaya, M. Li, L. Huang, and R. Poovendran, "AMOEBA: robust location privacy scheme for VANET," *IEEE Journal on Selected Areas in Communications*, vol. 25, no. 8, pp. 1569 –1589, Oct. 2007.

[55] Zhendong Ma, "Location privacy in vehicular communication systems: a measurement approach," Ph.D. dissertation, University of Ulm, Ulm, 2011.

[56] T. S. Heydt-Benjamin, H.-J. Chae, B. Defend, and K. Fu, "Privacy for public transportation," in *Privacy Enhancing Technologies*, 2006.

[57] E.-O. Blass, A. Kurmus, R. Molva, and T. Strufe, "PSP: private and secure payment with RFID," in *WPES*, 2009.

[58] F. Baldimtsi, G. Hinterwalder, A. Rupp, A. Lysyanskaya, C. Paar, and W. Burleson, "Pay as you go," in *HotPETs*, 2012.

[59] R. A. Popa, H. Balakrishnan, and A. Blumberg, "VPriv: protecting privacy in location-based vehicular services," in *USENIX Security Symposium*, 2009.

[60] J. Balasch, A. Rial, C. Troncoso, C. Geuens, B. Preneel, and I. Verbauwhede, "PrETP: privacy-preserving electronic toll pricing," in *19th USENIX Security Symposium*, 2010.

[61] S. Meiklejohn, K. Mowery, S. Checkoway, and H. Shacham, "The phantom tollbooth: Privacy-preserving electronic toll collection in the presence of driver collusion," in *20th USENIX Security Symposium*, 2011.

[62] Xihui Chen, G. Lenzini, S. Mauw, and J. Pang, "A group signature based electronic toll pricing system," in *ARES*, 2012.

[63] Chao Li, "Anonymous payment mechanisms for electric car infrastructure," Master's thesis, LU Leuven, Leuven, 2011.

[64] J. Camenisch, S. Hohenberger, and A. Lysyanskaya, "Compact e-cash," in *Advances in Cryptology - EUROCRYPT*, 2005.

[65] J. Liu, M. Au, W. Susilo, and J. Zhou, "Enhancing location privacy for electric vehicles (at the right time)," in *ESORICS*, 2012.

[66] M. H. Au, W. Susilo, and Y. Mu, "Constant-size dynamic k-TAA," in *Security and Cryptography for Networks*, 2006.

[67] M. Stegelmann and D. Kesdogan, "Design and evaluation of a privacy-preserving architecture for vehicle-to-grid interaction," in *EuroPKI*, 2012.

[68] A. Cavoukian, J. Polonetsky, and C. Wolf, "Smartprivacy for the smart grid: embedding privacy into the design of electricity conservation," *Identity in the Information Society*, vol. 3, pp. 275–294, 2010. [Online]. Available: http://dx.doi.org/10.1007/s12394-010-0046-y

[69] W. Kempton and J. Tomic, "Vehicle-to-grid power implementation: From stabilizing the grid to supporting large-scale renewable energy," *Journal of Power Sources*, vol. 144, no. 1, pp. 280–294, Jun. 2005. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0378775305000212

[70] J. Camenisch and E. Van Herreweghen, "Design and implementation of the idemix anonymous credential system," in *9th ACM Conference on Computer and Communications security*, 2002.

[71] J. Camenisch and A. Lysyanskaya, "An efficient system for non-transferable anonymous credentials with optional anonymity revocation," in *Advances in Cryptology - EUROCRYPT*, 2001.

[72] M. Ramsey, "Ford CEO: battery is third of electric car cost," *Wall Street Journal*, Apr. 2012. [Online]. Available: http://online.wsj.com/article/SB10001424052702304432704577350052534072994.html