Privacy-friendly Energy-metering via Homomorphic Encryption

Flavio D. Garcia and Bart Jacobs

Institute for Computing and Information Sciences,
Radboud University Nijmegen.
P.O. Box 9010, NL-6500 GL Nijmegen, The Netherlands.
{flaviog,bart}@cs.ru.nl

Abstract The first part of this paper discusses developments wrt. smart (electricity) meters (simply called *E*-meters) in general, with emphasis on security and privacy issues. The second part will be more technical and describes protocols for secure communication with *E*-meters and for fraud detection (leakage) in a privacy-preserving manner, using a combination of Paillier's additive homomorphic encryption and additive secret sharing.

Keywords: smart-metering, privacy, homomorphic encryption

1 Introduction

Many countries, for instance in Europe and North America, are currently undergoing changes in their electricity infrastructure, in which a better match between production and consumption is one of the goals. Accurate usage data is important for such a better match. So-called smart meters, or advanced meters [LGGG07], or *E*-meters, for consumers are a basic element in building a "smart grid" for electricity production and distribution. Frequent meter readings can be used to optimise the grid, but also reveal behavioural patterns, for instance about whether the inhabitants are at home, or at what time they get up or go to bed. Refined data analysis/mining over longer periods may reveal further information, for instance about the kind of devices that are being used, at which time, *etc.*

Privacy concerns are thus highly relevant in this context, and should be taken seriously by the utility sector. For instance, in April 2009 the Senate in the Netherlands has refused to pass a bill that made it compulsory for consumers to accept *E*-meters in their homes, precisely because of privacy concerns—and more generally, data protection concerns. This blocking of mandatory roll-out worked as wake-up call for the utility sector, at least in the Netherlands.

The issues of privacy, data protection and computer security are being addressed by various parties, see for instance the report [CPW09] by the Canadian information and privacy commissioner, or [NIS10] by NIST in the US (see also [EPI09]). Most of the emphasis in these documents lies on regulation via

standards, procedures, rules of conduct, auditing, independent oversight, *etc.* The emphasis in this paper will be on using technical means for achieving certain security and/or privacy goals, via privacy-protecting cryptographic techniques. In this way data minimalisation is enforced not only by design but also by implementation. The cryptographic techniques ensure that sufficient information is available to achieve certain goals, without revealing additional (privacy-sensitive) information. Specifically, Section 4 uses homomorphic (Paillier) encryption and additive secret sharing to make the total consumption readings visible at the neighbourhood level, without revealing *E*-meter readings at the household level. By comparing the total with the measurement of the actual consumption at the neighbourhood level, electricity leakage (via fraud) can be detected.

The first part of the paper discusses general issues in (electricity) metering and argues towards the inclusion of a trusted element, like a smart card, in *E*-meters. This is reflected in the slogan "power to the meter!". Such a trusted element provides secure storage of meter readings (like the traditional meter does via hardware protection), and basic cryptographic primitives based on public key cryptography, for authentication and secure communication. The protocols later on in the paper are based on the availability of such primitives. They demonstrate how basic cryptographic techniques can be used to achieve justifiable monitoring aims of grid operators without violating privacy of consumers.

In particular, Section 4 describes a protocol whereby data concentrators at the neighbourhood level can obtain sums of the measurements of all the connected customers (typically a few hundred) without learning the individual measurements. By comparing this sum with its own measurement of the consumed amount, it becomes clear how much energy leaks in this neighboorhood. These protocols may be run frequently, say every 15 minutes, without affecting privacy. In case serious leakage levels are found, additional means of investigation will have to used to detect the reason. How to do this is beyond the scope of the current paper.

In Section 5 appropriate security notions are introduced for the protocol from Section 4. In essence they say that an adversary should not be able to notice the swapping of the measurements of two customers. A sketch of a security proof is included for our protocol.

2 Background on smart metering

This section discusses the main players, the concerns and architecture for *E*-metering. We shall not go deeply into the *E*-metering set-up, and abstract for instance from the technique for communication with *E*-meters (GSM, power line communication, ...) and from the technique for the measurement of electricity consumption.

2.1 Stakeholders

The main stakeholders that we distinguish are:

- The electricity producer, *i.e.* the company that produces electricity and sells it to its customers. It needs accurate data about how much to produce at which moment, and how much (generation) capacity it needs to keep in reserve. Additionally, it needs cumulative usage data of individual customers for billing, on a monthly or bi-monthly basis.
- The grid operator, *i.e.* the company that controls the infrastructure for the distribution and transportation of electricity from producers to customers, and returns usage data to producers. In principle the metering can also be done by a separate party, but this is not what is assumed here for reasons of simplicity. Grid operators need accurate data about electricity flows and status information about essential grid components, in order to optimise their networks.
- The consumer of electricity, which is in the present setting a household consumer, and not a larger organisation (for which there are usually separate arrangements). European electricity regulation foresees 80% of consumers equipped with *E*-metering systems by 2020. Consumers must be regularly made aware of their energy consumption and its associated costs, in the hope that this leads to energy savings.

There are of course more stakeholders in this field, like regulators, (national) authorities, and *E*-meter producers. Their role will not be discussed here.

A fundamental question is: how much information do the operators and producers need to run their operations? For electricity producers this is relatively easy: they need cumulative and not continuous information for billing, and statistical information for usage patterns. This does not have serious privacy implications. Grid operators may need more information to efficiently run their network. From a privacy perspective, the question is whether they need usage information about individual households, or whether aggregated information from so-called substations at the neighbourhood level suffices. A stumbling block in the current debate is that no clear answers are given to this question. Therefor the operators seem to want all information, at the individual household level, with short intervals, down to quarter hour measurements, and they will see later how much they will actually use.

From a data protection and privacy perspective this attitude is clearly unacceptable. First, justifiable goals for data gathering must be clearly defined, and subsequently data minimization techniques must be applied to achieve these goals, with not more (identifiable) data than strictly necessary.

Experience in the Netherlands shows that grid operators find it difficult, and are thus reluctant, to define their goals other than in very generic terms (optimisation of their grids). However, the existing level of resistance forces them to take these issues more seriously.

2.2 Privacy concerns

Frequently measuring electricity consumption is privacy sensitive, because it reveals behavioural patterns that can be abused in various ways.

1. Daily measurements reveal any day whether a house is inhabited or not. It thus shows when the inhabitants are away for a weekend, or for a couple of weeks, on holidays. This information is relevant for burglars, for instance. Out-of-context storage of such measurement data in the servers of grid operators creates vulnerabilities, because the servers may be hacked, or system managers may be bribed or blackmailed into handing the data over to malicious outsiders.
2. More frequent, hourly or even quarter-hourly measurements, reveal even more information. Devout muslims get up at five in the morning for their first prayers, and can thus be singled-out, with some level of certainty (possibly in combination with names). Whether or not people are staying over in one-person flats may be noticeable. For instance, Figure 1 displays hourly measurements of electricity, water and gas of a particular home[1]. It seems that the inhabitants arrive at home at five in the afternoon, and that one (or more?) of them is taking a shower or bath at one o'clock at night. These measurement data are collected within private homes, "behind the front door" and inside "my home as my castle". They may make people exposed.
3. Long-term detailed insight in power consumption enables data mining and profiling in various ways (see [Har89]). For instance, the grid operators can observe certain patterns, like when the fridge switches on and how much electricity it uses. The operator can even observe if such a fridge becomes old (less efficient) and needs to be replaced soon. This information can be used, for instance, for targeted (fridge) advertisements, listing a particular brand of fridges only—for which the grid operator gets a commission with each sell. Consumers may view this positively as a service or negatively as a form of intrusion.

A recent report [CK08] commissioned by the consumer organisation in the Netherlands, argues that frequent reading of *E*-meters is problematic from a legal perspective, specifically because it violates article 8 on Privacy of the European Convention on Human Rights: a pressing need in a democratic society to force people to deliver privacy sensitive usage data is lacking. The setup which is foreseen may thus be challenged at some stage in (European) court. Hence a long-term perspective, building on trust and societal acceptance, is needed.

An additional sensitive issue is that remote reduction, or even shutdown, of electricity supply is foreseen, for instance in case of nonpayement. This creates a huge denial of service (DOS) risk, not only for individual customers, but also for national security (in case of cyber warfare, or as force multiplier in a terrorist attack)[2].

[1] The homeowner(s) voluntarily put these data on the web, see `bwired.nl`.

[2] It thus makes sense to restrict this remote shutdown to a certain minimal level only, so that one can still switch on the light but not the whirlpool bath. This can be

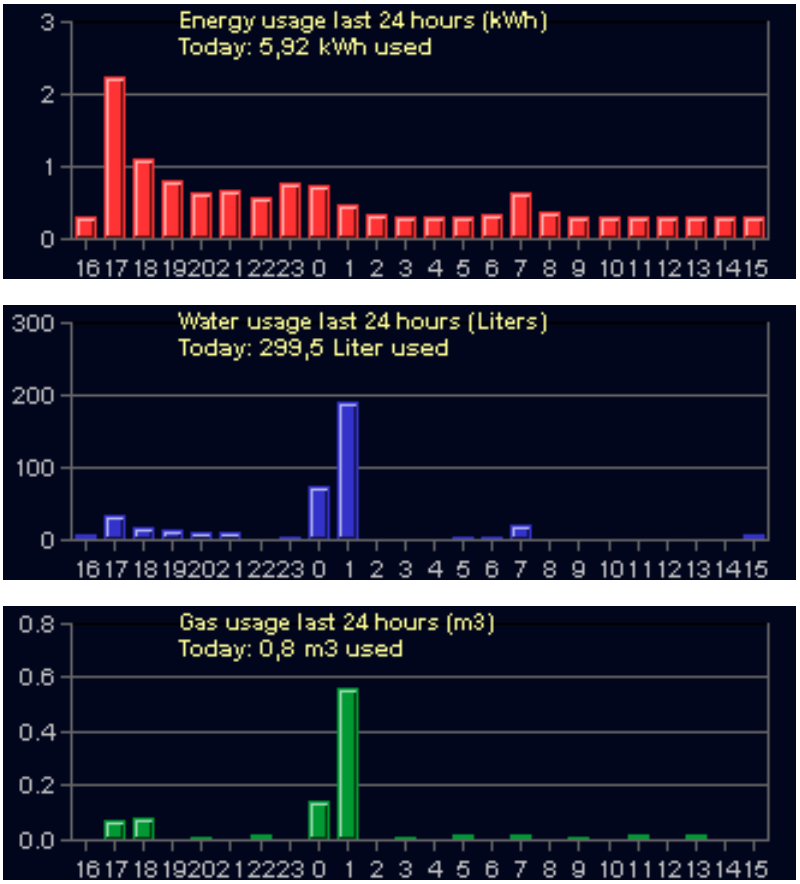

Figure 1. Example hourly measurements of electricity, water and gas, from `bwired.nl`

2.3 Centralised trust

Traditional (legacy) electricity meters have a physical counter that moves forward due to a metallic disc that turns with a speed proportional to the electricity consumption. This process takes place within a sealed container, so that tampering is not so easy and can be detected. It involves local storage of cumulative usage levels, in a way that is fairly reliable. Such domestic meters may last for decades. The hardware protection works in two directions: it prevents that customers (easily) manipulate (esp. decrease) the meter readings, so that they pay less than they should. But it also prevents that electricity suppliers (easily) manipulate (esp. increase) the meter readings so that customers pay more than they should. Additionally it protects against other possibly malicious actors. This set-up with protected local storage is essential for guaranteeing a reasonable level of trust, in two directions.

The focus of the current generation of *E*-meters is on protection against manipulation by customers. No (technical) protection is foreseen against manipulation by grid operators, since they can remotely update not only crucial values like timers and cryptographic keys, but also install new software and thus completely replace the functionality of the meters whenever they like. Grid operators thus need to be universally trusted. Customers are protected only by procedural guidelines, audits, codes of behaviour ("we do not read your meter surreptitiously"). Furthermore, the measurement data are no longer only locally securely stored, in-context, but they are stored (also) centrally, out-of-context in the database of the operator.

This change in responsibilities and in the balance of power is remarkable. Whether it is in the long term interest of the operators (and the utility sector at large) remains to be seen. When people are forced to trust a single party, this trust may suddenly collapse, like for voting machines [JP09]. Abuse will get media attention and the premiss of universal trust in the operator will then be questioned. Also, customers may contest their bills in court. If such a customer claims his bill is incorrect or even fabricated by the grid operator (or by someone else) because of software malfunctioning, the grid operator is in such a set-up in a weak position to defend itself. After all, the grip operator controls all software, timing, storage, and cryptographic keys of the *E*-meter.

2.4 Secure authentication and local storage: "power to the meter"

Usually in the security architecture of a distributed system one identifies the different islands of trust, and provides each of them with their own trusted computing base (TCB), and with secure communication lines between them, going across security boundaries through untrusted territories. The current design does not have such a structure. Instead, there is only one party that is universally trusted (the grid operator) and has total control over the others (notably the

achieved physically by having two wires, and restricting the shutdown functionality to only one of them.

E-meters and substations at the neighbourhood level), since it has the "divine" power to (remotely) replace the software and cryptographic keys of the others.

A more robust and trustworthy design gives the *E*-meters (and probably also the substations) a certain level of autonomy via trusted elements, providing secure storage and autonomous cryptographic functionality. These trusted elements should lie outside the reach of the operators. In particular, it should not be possible to remotely change their software or change their cryptographic keys. The software of the device in which the trusted element resides may then be updated remotely, because the device is not part of the trusted computing base[3]. A similar, but more elaborate form of local autonomy, is proposed in [LGGG07] via the Trusted Platform Module (TPM), hypervisors, and several separate Virtual Machines (VMs).

Via such trusted elements, like smart cards or secure USB sticks, *E*-meters can digitally sign the messages they send, including the meter readings. This provides confidentiality and integrity, but also non-repudiation, which should be the basis of conflict resolution, see Section 3.

In a rather predictable future scenario electric cars will be used more extensively. A domestic *E*-meter could be provided with several such USB sticks for mobile electricity consumption. When visiting a friend by (electric) car, charging the batteries over there can be billed to my own account via such a USB stick for secure authentication. It thus makes sense to build secure authentication deep into the architecture of *E*-meters, certainly if this infrastructure is meant to last for decades[4].

In the remainder of this paper we shall thus assume that *E*-meters have secure elements providing a trusted computing base with secure storage and basic cryptographic functionality, including public key cryptography. We assume that the software of these trusted elements cannot be changed remotely by the operators. Also, the private keys of, or generated by, these trusted elements are inaccessible from the outside. They do have a number of certificates (with public keys), for instance of the grid operator, of a number of electricity producers, and possibly of additional service providers. New certificates may be sent to the trusted element, provided with appropriate signatures, for instance after expiry of the old ones, or when new parties arrive on the market. Ideally, these trusted elements also have their own clock and power supply, for instance embedded in a USB stick. In case of a major security break down these trusted elements will have to be replaced (physically).

Current *E*-meters, as far we know, do not have such separate trusted elements and rely mostly on symmetric key cryptography (if any). The security level that they provide is limited, see for instance [KR08]. The *E*-meter market is not very mature yet, making early large scale roll-out risky.

[3] Admittedly, this architecture is more complicated than sketched here, because the trusted computing base must also include the measurement sensor and a clock.

[4] Such non-domestic electricity consumption introduces additional location-sensitive privacy issues which form a topic on its own.

3 Basic protocols

This section sketches some protocols for basic communication with *E*-meters, using elementary cryptographic operations. They ensure that messages are authenticated, fresh (to prevent replay), and confidential. Moreover, they provide integrity protection and non-repudiation. These protocols are fairly obvious but are included to demonstrate how basic cryptographic primitives can secure the communication and provide authenticity. Their implementation with existing technology is unproblematic. For instance, we have our own prototype implementation using cheap Java-enabled smart cards.

This protocol involves three parties:

- the smart-meter $\mathcal{M}$;
- the grid operator *GO*;
- the supplier $\mathcal{S}$.

Notation We write $\{\, m \,\}_{\mathcal{A}}$ to denote the encryption of message m under $\mathcal{A}$'s public key; $\mathcal{K}$ is the key generation algorithm; $[\, m \,]_{\mathcal{A}}$ denotes the signature produced by $\mathcal{A}$ on the message m.

Initially, $\mathcal{M}$ holds the public key of *GO* and the key of the certification authority *CA*. The grid operator *GO* might initiate the interactive protocols **set_supplier** and **switch_power** with $\mathcal{M}$. In contrast to the current setup where the grid operator can access the meter readings at will, we propose a setting where grid operators can set a reading policy, indicating who is the energy supplier and how often the meter is supposed to report the meter readings for billing purposes. This time period P, typically 2 months, can be shown to the consumers on the meter's display, and then it is the meter itself who initiates the **meter_report** protocol.

Concretely, in the **set_supplier** protocol, the grid operator *GO* says hello, I want to set a new supplier. Then the meter sends a challenge nonce n in order to ensure freshness. Then *GO* sends an encrypted and signed message containing the new policy, *i.e.*, the identity of the new supplier and its public key, the time period P for the reports and a time-stamp ts determining when the new supplier takes over.

set_supplier:

$GO \to \mathcal{M}$: hi, init set_supplier

$\mathcal{M} \to GO$: nonce n

$GO \to \mathcal{M} : \{\, [\, \text{set\_supplier}, \mathcal{M}, n, \mathcal{S}, \text{pk}_{\mathcal{S}}, \text{ts}, P \,]_{GO} \,\}_{\mathcal{M}}$

In emergency situations or in case of nonpayment, there is the requirement that grid operators can unplug a household from the grid or there is also the possibility of a partial disconnect where the meter allows only a few kW/h for basic needs. The following **switch_power** protocol implements this functionality, where power $\in [0, 1]$ represents the permitted consumption, being 0 totaly unplugged and 1 fully operational.

switch_power:

$GO \rightarrow \mathcal{M}$: hi, init switch_power

$\mathcal{M} \rightarrow GO$: nonce n

$GO \rightarrow \mathcal{M} : \{\, [\,\text{switch\_power}, \mathcal{M}, n, \text{ts}, \text{power}\,]_{GO}\,\}_{\mathcal{M}}$

Starting from the time ts from the **set_supplier** protocol and for every period of time P, the meter $\mathcal{M}$ will report the meter readings to the supplier $\mathcal{S}$. This message, in fact, is being relayed by the grid operator to the supplier, but we abstract from that since end-to-end encryption is used.

meter_report:

$\mathcal{M} \rightarrow \mathcal{S} : \{\, [\,\mathcal{M}, \text{time, meter readings}\,]_{\mathcal{M}}\,\}_{\mathcal{S}}$

These protocols are sufficient for billing purposes and, provided that the time period P is large enough, the privacy sensitive information revealed is minimal.

There is one issue though, for leakage or fraud detection and smart-grid optimization, grid operators claim that they need much more frequent readings, more in the order of every 15 minutes. On the positive side, for this tasks they do not need the specific readings of each meter, but it is enough to know the aggregated consumption at block or neighborhood level. Section 4 describes a protocol that achieves such goals in a privacy-friendly manner.

4 The no-leakage protocol

We assume a local substation *SSt* that is connected to several customer meter devices $\mathcal{M}_1, \ldots, \mathcal{M}_N$, as in Figure 2. Typically N is in the order of a few hundred. The number of meter devices connected to *SSt* may change over time, due to addition or removal of meters.

The *SSt* supplies electricity / gas / water, to the $\mathcal{M}_i$, and measures these total supplies m_{SSt} at regular intervals, for instance every 15 minutes. The $\mathcal{M}_i$ have their own (regular) measurements m_i, and report these measurements back to the *SSt*.

The goal of the no-leakage protocol is to learn the aggregated energy consumption of N consumers (think of a neighborhood) without revealing any information about the individual consumption of the users, even when the data concentrator is malicious, *i.e.*, does not necessarily follow the protocol.

The security of this protocol relies on the assumption that at least two out of N users in the neighborhood are uncorrupted, *i.e.*, they behave according to the protocol specification. More precisely, we assume that there is a trusted certification authority that issues certificates and that the adversary is unable to obtain a large number of public key certificates for which he knows the private key. This assumption seems unavoidable since it is inherent in the problem that when the adversary knows the consumption of all-but-one consumers then she can trivially learn the consumption of the last consumer by simply running the protocol and subtracting.

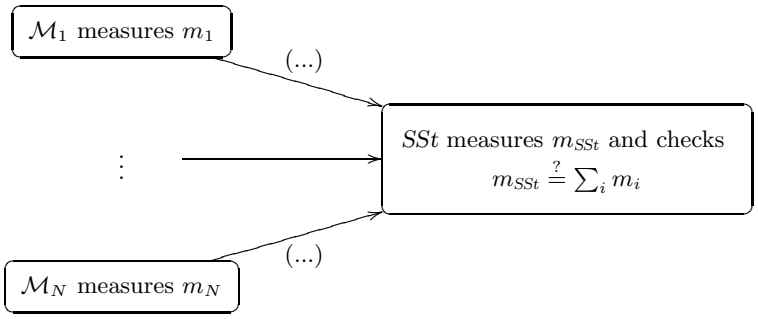

Figure 2. Neighboorhood set-up, with supplies to the $\mathcal{M}_i$ going via SSt, and protected measurements m_i sent back at regular intervals. The total supply m_{SSt} measured by SSt should equal the sum of the m_i, at each interval.

Let $\{\cdot\}.$ be an IND-CPA secure, additively homomorphic encryption scheme satisfying $\{m_1\}_k \cdot \{m_2\}_k = \{m_1 + m_2\}_k$ like Paillier [Pai99]. Assume that each meter $\mathcal{M}_i$ has a public key certificate $cert_{\mathcal{M}_i}$ of his public key pk_i, and he also has knowledge of the corresponding private (decryption) key.

The no-leakage protocol is depicted in Figure 3. The data concentrator initiates the protocol by sending the public key certificates of all users in the neighborhood. If the number of certificates is smaller than the minimum neighborhood size allowed N_{min}, then the participants abort the protocol. This prevent information leakage when the neighborhood is too small, ultimately of size one. Then,

$SSt \longrightarrow \mathcal{M}_i$: no-leakage; $cert_{\mathcal{M}_1}, \ldots, cert_{\mathcal{M}_N}$

$\mathcal{M}_i \longrightarrow SSt$: $y_{i1}, \ldots, y_{ii-1}, y_{ii+1}, \ldots, y_{iN}$

where $\mathcal{M}_i$ picks random numbers $a_{i1}, \ldots, a_{iN}$ s.t. $m_i = \sum_j a_{ij} \mod n$

and sets $y_{ij} := \{a_{ij}\}_{\mathrm{pk}_j}$ for $j = 1, \ldots, i-1, i+1, \ldots, N$.

If $N < N_{\min}$ then $\mathcal{M}_i$ aborts.

$SSt \longrightarrow \mathcal{M}_i$: $\prod_{j \neq i} y_{ji} = \{\sum_{j \neq i} a_{ji}\}_{\mathrm{pk}_i}$ (due to the homomorphic property of $\{\cdot\}.$)

$\mathcal{M}_i \longrightarrow SSt$: $\sum_{j \neq i} a_{ji} + a_{ii} = \sum_j a_{ji} \mod n$

SSt sets $m := \sum_i \sum_j a_{ji} \mod n$.

Figure 3. no-leakage protocol

each $\mathcal{M}_i$ prepares N shares $a_{i1}, \ldots, a_{iN}$ of its measurement m_i s.t. $m_i = \sum_j a_{ij}$ mod n, for a large n. $\mathcal{M}_i$ encrypts each of these shares a_{ij} with the public key pk_j of user $\mathcal{M}_j$ and sends them back to SSt, except for the share a_{ii}, which is simply remembered locally by $\mathcal{M}_i$. Next, SSt multiplies all $N-1$ ciphertexts

intended to user $\mathcal{M}_i$ and sends the resulting ciphertext to him to decrypt. Due to the aforementioned homomorphic property of the cipher, this equals the sum of these $N-1$ shares. Next, $\mathcal{M}_i$ decrypts the received ciphertext, adds a_{ii} to the plaintext and sends back the result to *SSt*. This later addition of a_{ii} results crucial to the security (and soundness) of the protocol. Finally, *SSt* collects the contributions from all users and adds them to obtain the aggregated consumption m which can be compared to m_{SSt}.

5 Security notions

This section elaborates precise security notions for metering protocols. We first recall the standard IND-CPA security for encryption schemes and then we introduce two new security notions for metering protocols: correctness and no-leakage.

Definition 5.1 (IND-CPA-Game)**.**

$\boldsymbol{IND\text{-}CPA\text{-}Game}_{\Pi,\mathcal{A}}(\eta)$:
$(sk, pk) \leftarrow \mathcal{K}(1^\eta)$
$p_0, p_1 \leftarrow \mathcal{A}_0(pk)$
$b \leftarrow \{0,1\}$
$b' \leftarrow \mathcal{A}_1(\{\, p_b \,\}_{pk})$
winif *if* $b = b'$.

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

Definition 5.2 (IND-CPA)**.** *An encryption scheme* Π *is said to be* IND-CPA *secure if for all probabilistic polynomial-time adversaries* $\mathcal{A} = (\mathcal{A}_0, \mathcal{A}_1)$

$$\mathbb{P}[\boldsymbol{IND\text{-}CPA\text{-}Game}_{\Pi,\mathcal{A}}(\eta)] - 1/2$$

is a negligible function of η.

Definition 5.3 (correctness)**.** *A protocol* Π *is said to be correct if it indeed outputs the aggregated consumption of all* N *participants, i.e., a value* $m = \sum_{i=0}^{N} m_i$.

Next we want to define what does it mean for a protocol to be non-leaking. We will do so in the style of what IND-CCA is for an encryption scheme. For that we first define an indistinguishability game and then we concretely define no-leakage. The intuition behind the definition is that if an adversary cannot even distinguish the swapping of the consumptions of two arbitrary users, then the protocol does not reveal information about the individual consumptions of the users.

The game proceeds as follows. First, the key generation algorithm is invoked to create a public/private key pair (sk, pk) for the certification authority *CA* and

for each meter. Then, the certification authority CA outputs the corresponding public key certificates and these are given to the adversary $\mathcal{A}_0$, together with the public key pk_{CA} of the CA. At this point the adversary is able to query the corruption oracle $\mathcal{O}$ in order to retrieve the private keys of a number of meters. At some point the adversary $\mathcal{A}_0$ stops and outputs two uncorrupted target meters $\mathcal{M}_0^\star$ and $\mathcal{M}_1^\star$ and two challenge consumption measurements m_0 and m_1. Then, the environment chooses a random bit b which determines a permutation of m_0 and m_1. Then, $\mathcal{A}_1$ can interact with the challenge meters and try to learn information about their consumption. At any time $\mathcal{A}_1$ might query the corruption oracle, but the restriction over the target meters still apply. Eventually $\mathcal{A}_1$ stops and outputs a guess b' for the bit b. We say that the adversary wins the game if $b = b'$.

Definition 5.4 (No-Leakage-Game)**.**

$$
\begin{array}{l}
\boldsymbol{No\text{-}Leakage\text{-}Game}_{\Pi,\mathcal{A}}(\eta) \ : \\
\quad (sk_{CA}, pk_{CA}) \leftarrow \mathcal{K}(1^\eta) \\
\quad (sk_0, pk_0) \ldots (sk_N, pk_N) \leftarrow \mathcal{K}(1^\eta) \\
\quad cert_0 \ldots cert_N \leftarrow CA(sk_{CA}, pk_0 \ldots pk_N) \\
\quad \mathcal{M}_0^\star, \mathcal{M}_1^\star, m_0, m_1 \leftarrow \mathcal{A}_0^{\mathcal{O}}(pk_{CA}, cert_0 \ldots cert_N) \\
\quad b \leftarrow \{0,1\} \\
\quad b' \leftarrow \mathcal{A}_1^{\mathcal{O}}(\mathcal{M}_0^\star(sk_0^\star, m_b), \mathcal{M}_1^\star(sk_1^\star, m_{1-b})) \\
\quad \boldsymbol{winif}\ \textit{if}\ b = b'.
\end{array}
$$

were the adversary $\mathcal{A}$ has access to a corrupting oracle $\mathcal{O}$ that on input the identity of a meter $\mathcal{M}_i$ returns its corresponding private key sk_i. The target meters $\mathcal{M}_0^\star$ and $\mathcal{M}_1^\star$ must be uncorrupted, which means that no $\mathcal{O}(\mathcal{M}_{\{0,1\}}^\star)$ query is made. Adversaries implicitly pass state i.e., from $\mathcal{A}_0$ to $\mathcal{A}_1$.

Definition 5.5 (No-Leakage)**.** *A protocol Π is said to be* non-leaking *if for all probabilistic polynomial-time adversaries $\mathcal{A} = (\mathcal{A}_0, \mathcal{A}_1)$*

$$\mathbb{P}[\boldsymbol{No\text{-}Leakage\text{-}Game}_{\Pi,\mathcal{A}}(\eta)] - 1/2$$

is a negligible function of η.

6 Security analysis

This section shows correctness and no-leakage properties of the protocol proposed in Section 4.

Theorem 6.1. *The protocol depicted in Fig. 3 is correct.*

Proof The proof of correctness is trivial, observe that

$$m = \sum_i \sum_j a_{ji} = \sum_i \sum_j a_{ij} = \sum_i m_i. \qquad \square$$

Theorem 6.2. *The protocol depicted in Fig. 3 is non-leaking.*

Proof Assume that there is an adversary $\mathcal{B}$ that wins the **No-Leakage-Game** with probability significantly larger than 1/2. Then we build the following adversary $\mathcal{A}$ against the IND-CPA security of the encryption scheme. For simplicity of the exposition and without loose of generality, assume that $N = 2$. If the no-leakage property holds for $N = 2$ then it holds for $N > 2$.

The adversary $\mathcal{A}$ will first simulate the environment for $\mathcal{B}$, this is, it will create a public key pair for the *CA* and a public key pair $(\mathrm{pk}_2, \mathrm{sk}_2)$ for $\mathcal{M}_2$ by calling $\mathcal{K}$. The public key pair $(\mathrm{pk}_1, \mathrm{sk}_1)$ of $\mathcal{M}_1$ will not be generated by $\mathcal{A}$ but the challenge public key pk from the INC-CPA game will be used instead of pk_1. Following the structure of the **No-Leakage-Game**, $\mathcal{A}$ will create the corresponding certificates $cert_{\mathcal{M}_1}$ and $cert_{\mathcal{M}_2}$. Then it calls $\mathcal{B}_0$ which will eventually output two target meters and two consumption measurements m_0 and m_1. Assume w.l.o.g that it outputs $\mathcal{M}_1$ and $\mathcal{M}_2$. Next $\mathcal{B}_1$ is called.

If at any point of the simulation $\mathcal{B}$ initiates a non-leakage protocol, then $\mathcal{A}$ will proceed as the protocol indicates choosing random $a_{11} + a_{12} = m_1$ and $a_{21} + a_{22} = m_2$. Then, instead of sending $y_{21} := \{\, a_{21} \,\}_{K_1}$ it will send $p_0 := a_{21}$ and $p_1 := a_{12}$ as challenge plaintexts for the IND-CPA game and it will get the challenge ciphertext $\{\, p_b \,\}_{k_1}$ in return, for a random bit b. $\mathcal{A}$ will choose a random bit $t \leftarrow \{0, 1\}$ of its own and set $y_{21} := \{\, p_b \,\}_{k_1}$ and $y_{12} := \{\, p_t \,\}_{k_2}$. When $t = 0$ it will also swap the values of a_{11} and a_{22} to keep a consistent protocol run. Observe that due to the later addition of a_{ii}, this does not affect $\mathcal{B}$'s view of the protocol. For the rest of the protocol $\mathcal{A}$ follows the protocol description. At some point $\mathcal{B}$ stops and outputs a guess b'. Then $\mathcal{A}$ also finishes and outputs the same guess b'. Note that, when $t = 1 - b$, $\mathcal{A}$ has the same distinguishing advantage than $\mathcal{B}$, and this happens with probability $\frac{1}{2}$. $\square$

7 Conclusions

This paper discussed several privacy issues in the current smart metering infrastructure. We conclude that this structure has to be rethought in order to replace a unilateral trust assumption by a more multilateral architecture where *E*-meters have a trusted component and enjoy a certain level of autonomy. A

trustworthy system should provide guarantees about the measurements for both grid operators *and* consumers. We have shown how to realise several tasks like billing, grid optimization and notably leakage detection in a privacy-friendly manner. The protocols proposed here are practical and can be straightforwardly implemented using inexpensive smart cards.

There is still much more research to be done in this area, but we hope that the concerns raised here will ignite fruitful discussions in an application area that this not yet up-to-date with the state of the art in cryptography.

References

[CK08] C. Cuijpers and B.-J. Koops. Het wetsvoorstel 'slimme meters': een privacytoets op basis van art. 8 EVRM. Technical report, Tilburg University, oct. 2008. Report (in Dutch).

[CPW09] A. Cavoukian, J. Polonetsky, and C. Wolf. SmartPrivacy for the Smart Grid: Embedding Privacy into the Design of Electricity Conservation, 17 Nov 2009. `www.ipc.on.ca/images/Resources/pbd-smartpriv-smartgrid.pdf`.

[EPI09] EPIC. Comments of the Electronic Privacy and Information Center on the NIST Smart Grid Standards, 1 Dec 2009. `http://epic.org/privacy/smartgrid/EPIC_Smart_Grid-Cybersecurity_12-01-09.2.pdf`.

[Har89] G. Hart. Residential energy monitoring and computerized surveillance via utility power flows. *IEEE Technology and Society Magazine*, 8(2):12–16, 1989.

[JP09] B. Jacobs and W. Pieters. Electronic voting in the netherlands: From early adoption to early abolishment. In A. Aldini, G.Barthe, and Gorrieri, editors, *Foundations of Security Analysis and Design V: FOSAD 2007/2008/2009 Tutorial Lectures*, number 5705 in Lect. Notes Comp. Sci., pages 121–144. Springer, Berlin, 2009.

[KR08] S. Keemink and B. Roos. Security Analysis of Dutch Smart Metering Systems. Technical report, Amsterdam: UvA, 7 July 2008. https://www.os3.nl/2007-2008/students/bart_roos/rp2.

[LGGG07] M. LeMay, G. Gross, C. Gunter, and S. Garg. Unified architecture for large-scale attested metering. In *HICSS '07: Proceedings of the 40th Annual Hawaii International Conference on System Sciences*, pages 115–125. IEEE Computer Society, Washington, DC, USA, 2007.

[NIS10] NIST. Smart Grid Cyber Security Strategy and Requirements, 2 Feb 2010. `http://www.itl.nist.gov/div893/csrc/publications/drafts/nistir-7628/draft-nistir-7628_2nd-public-draft.pdf`.

[Pai99] P. Paillier. Public-key cryptosystems based on composite degree residuosity classes. In Jacques Stern, editor, *Advances in Cryptology (EUROCRYPT'99)*, volume 1592 of *Lecture Notes in Computer Science*, chapter 16, pages 223–238. Springer Berlin Heidelberg, Berlin, Heidelberg, 1999.