TRANSACTIONS ON EMERGING TELECOMMUNICATIONS TECHNOLOGIES
Trans. Emerging Tel. Tech. (2013)
Published online in Wiley Online Library (wileyonlinelibrary.com). DOI: 10.1002/ett.2760

RESEARCH ARTICLE

A protocol for metering data pseudonymization in smart grids†

Cristina Rottondi*, Giulia Mauri and Giacomo Verticale

Dipartimento di Elettronica, Informazione e Bioingegneria, Politecnico di Milano, Piazza Leonardo da Vinci, 32, Milano, Italy

ABSTRACT

A tradeoff between data collection needs and user privacy is of paramount importance in the Smart Grid. This paper proposes a pseudonymization protocol for data gathered by the Smart Metres, which relies on a network infrastructure and a dedicated set of nodes, called privacy preserving nodes. The network privacy is enforced by a separation of duties; the privacy preserving nodes perform data pseudonymization without having access to the measurements, which are masked by means of a secret sharing scheme, while the entities accessing the data recover and relate the plain measurements generated by the same metre along a time window of finite duration but have no access to the metre identities. The paper also provides an evaluation of the security and of the performance of the protocol, comparing it to the two alternative encryption techniques, which mask the measurements by means of the Chaum mixing scheme or of an identity-based proxy re-encryption scheme. Copyright © 2013 John Wiley & Sons, Ltd.

***Correspondence**
Cristina Rottondi, Dipartimento di Elettronica, Informazione e Bioingegneria, Politecnico di Milano, Piazza Leonardo da Vinci, 32, Milano, Italy.
E-mail: rottondi@elet.polimi.it

†A preliminary version of this paper appears in C. Rottondi, G. Mauri, and G. Verticale, "A Data Pseudonymization Protocol for Smart Grids", in *IEEE GEENCOM, Online Green Communications Conference*, September 2012.

Received 27 May 2013; Revised 26 September 2013; Accepted 14 October 2013

1. INTRODUCTION

Many countries are modernising their power grid into a more stable and efficient 'Smart Grid', which is expected to be reliable, scalable, manageable and extensible but is also secure, interoperable, and cost-effective. Therefore, the new grid is a very complex ecosystem involving both information technology and power grid operations and governance.

The Smart Grid can capture and analyse data regarding power usage and generation in near-real time, providing forecasts and recommendations to the customers regarding the optimization of their consumption. Such a system presents many privacy and cyber-security challenges. In fact, the Smart Metres collect energy consumption data with high frequency, improving the quality of the information available to enhance electricity provision, adding value to services for customers and improving billing. However, this can also result in a violation of people's privacy, because users' personal habits and customs can be inferred by analysing energy consumption data gathered by the metres. Therefore, numerous governmental authorities and standardisation bodies have emanated rules and restrictions about the treatment and diffusion of metre readings; for example, National Institute of Standards and Technology [1] mandates that, unless strictly necessary, metering data should be anonymized in order to prevent utilities and third parties from linking the collected information to the identity of the customers that generated them. A recent act of the USA Committee on Homeland Security [2] imposes that procedures for the anonymization of cyber-information must be defined in order to make such information available to external parties, for example, for academic research or actuarial purposes. Depending on the specific contexts and applications, data anonymization can be achieved through different approaches [3], ranging from generalisation (where information is coarsened into representative sets) to perturbation (where data are polluted by means of noise addition), pseudonymization (which replaces the individuals' true identities with pseudonyms) and aggregation (which releases cumulative data computed on the information provided by multiple individuals, so that the contribution of a single entity is no longer identifiable in the aggregated data).

This paper proposes a metering data pseudonymization protocol (first introduced in [4]) that allows multiple external entities (such as utilities, third parties and service providers) to obtain disaggregated data generated by the Smart Metres. Data maintain their temporal sequentiality along a time window of finite duration, but the protocol does not allow the association between the data and the identity of the metre that generated them. To do so, we rely on a privacy-preserving infrastructure that introduces a set of privacy preserving nodes (PPNs) in the Smart Grid architecture. The infrastructure shares similarities with the one we defined in [5] for anonymization through aggregation. In fact, the same set of PPNs could provide both aggregation, as described in [5], and pseudonymization, as described in this paper. Note that the PPNs allow a considerable reduction of the computational effort required to the energy metres, which are usually resource-constrained low cost devices, with limited computational capabilities. The PPNs can be operated in a centralised fashion by independent parties or regulation authorities, as envisioned by different research and standardisation bodies. For example, a recent proposal of the California Public Utilities Commission [6] speculates the realisation of energy data centres aimed at the collection and dissemination of aggregated and/or anonymized energy consumption data and run by governmental entities. Collecting centres gathering data from local units deployed at household level in a global system for mobile Global System for Mobile Communications (GSM)/general packet radio service (GPRS) based automatic metering infrastructure (AMI) have been demonstrated by several utilities in the Shandong province of China [7]. While such data centres are assumed to be fully trusted, our proposed architecture ensures no violation of the customers' privacy even in the presence of 'honest-but-curious' collectors, thus not requiring full trustability. However, the functionalities of the PPNs could also be performed by nodes already existing in the Smart Grid scenario; a hierarchical smart metering system architecture providing a non-trusted k-anonymity service based on gateways located at the customers' premises and mandated by the German Federal Office for Information Security [8] has been discussed in [9]. A fully distributed peer-to-peer network for AMI using Smart Metres realised with off-the-shelf hardware and relying on existing communication infrastructure has recently been proposed by a German technology provider [10].

Although customer pseudonymized data cannot be used for billing purposes, nevertheless, their release would be highly beneficial for numerous entities with a wide variety of scopes. For example, vendors and societies conducting target marketing should not have access to user-related data but only to anonymized measurements, because energy load profiles could for example be used to track the usage of particular electrical appliances [11]. Another relevant example is the analysis of the anonymized energy usage patterns aimed at monitoring the distributed generation capabilities of the customers, the rate impact within/across climate zones and other relevant parameters to evaluate the efficiency of the Smart Grid [6].

It is worth noting that our privacy-preserving infrastructure can also support spatial and/or temporal aggregation of metering data according to the needs expressed by the external entities, which we investigated in [5]. Moreover, as it will be detailed in the remainder of the paper, our framework can be easily integrated with data perturbation and obfuscation techniques. Therefore, we can state that our proposed infrastructure can combine three different methods to perform data anonymization, in order to increase the resiliency of the system to different categories of de-anonymization attacks to smart metering data (as described, e.g. in [12]). This paper gives the following contributions:

(1) Defines a set of security properties, which capture the needs of the Smart Grid scenario.
(2) Proposes a pseudonymization cryptosystem for frequent re-pseudonymization and multiple external entities.
(3) Describes a network architecture and a communication protocol for pseudonymizing metering data and analyses how they satisfy the stated security properties.
(4) Compares different cryptographic approaches for preventing the network from accessing the metering data and shows that the Shamir secret sharing (SSS) scheme is the only one compatible with real-time operations.

We compare our protocol with two alternative very popular encryption techniques, the Chaum mixing scheme [13–16] and the identity based proxy re-encryption (IB-PRE) scheme [17–23], which have been proposed by the research community and applied in numerous scenarios, ranging from web browsing to data storage and sharing.

The remainder of the paper is structured as follows. Section 2 provides an overall view about data pseudonymization and anonymization in various contexts of communication networks, focusing on the Smart Grid scenario. Section 3 recalls some background notions. Section 4 introduces the pseudonymization framework and its possible deployment in the Smart Metering system, while the pseudonymization cryptosystem and its implementation are discussed in Section 5. Section 6 describes the communication protocol relying on SSS scheme and compares it to two alternative cryptographic schemes, while Section 7 discusses the security guarantees the protocol achieves. The security assessment and performance evaluation are provided in Section 8. Concluding remarks are left for the final Section.

2. RELATED WORK

The problem of metering data pseudonymization in the context of Smart Grids has recently attracted the interest of several researchers. Efthimiou and Kalogridis [24]

describe a method for the anonymization of electrical metering data sent by Smart Metres. They propose to separate high frequency and low frequency data and to assign an identity to each set of measurements: the high frequency identification (ID) is anonymous, while the low frequency ID is attributable. The association between the two IDs is prevented by inserting long random intervals during the system setup. This solution has the drawback of requiring a long setup time and of hard-coding the IDs in the Smart Metre itself.

Jawurek *et al.* [12] develop two attacks to the privacy of pseudonymized consumption traces; the first is used to link an identity to a consumption trace by anomaly correlation, while the second links different pseudonyms of a customer by identifying common patterns in electricity consumption. The authors also analyse three countermeasures based on data aggregation, frequent re-pseudonymization and privacy preserving techniques and provide numerical values for the correct tuning of the time aggregation and re-pseudonymization windows. In particular, they state that raising the time aggregation windows from 3 to 24 h causes a decrease in the accuracy achieved through behavioural patterns linking from 70% to 4%, while re-pseudonymization must be performed every one or very few days to obtain an accuracy below 50% (20 days long intervals lead to accuracy levels above 80%). Our protocol allows the choice of an arbitrarily low re-pseudonymization time window.

A privacy preserving protocol is presented by Rial and Danezis [25]. In this scheme, the metre outputs certified readings of measurements using cryptography; the user combines those readings with a certified tariff policy to produce the final bill. A zero-knowledge protocol ensures the correctness of the bill. The proposed protocol guarantees integrity and privacy and can perform the secure computation of a generic additive function, while our protocol does not perform any elaboration on the collected data but addresses the issue of replacing the identities of the subjects generating the measurements with unattributable pseudonyms.

Stegelmann and Kesdogan [9] analyse the issues related to anonymity and pseudonymity within the German BSI's Protection Profile. The authors identify several problems and propose GridPriv, an architecture with a non-trusted k-anonymity service that allows to overcome the challenges. In particular, they consider churning attacks that a service provider can perform to determine an anonymity set, that is, a set of Gateways, which can be the data's originator. Their architecture ensures anonymity within a set of a certain size k, whereas our protocol guarantees pseudonymity for all the Smart Metres generating the data.

Privacy protection is an important topic also in other contexts, from mobile ad hoc networks to radio-frequency identification (RFID) systems and health-care. Public-key based solutions have been proposed to guarantee communication anonymity, which means that the sender's and receiver's identities are hidden to external observers. Zhang *et al.* propose in [26] a pairing-based anonymous on-demand routing protocol; in this approach, a trust authority administrates the anonymous communication system by providing each node with a sufficiently large set of collision resistant pseudonyms, so that each node can dynamically change its pseudonym and communicate the set of system parameters to each anonymous user. Although the protocol guarantees sender anonymity, receiver anonymity and relationship anonymity, the communications are not anonymous to the trust authority. To solve this problem, Huang proposes in [27] pairing-based encryption/decryption, key exchange, blind certificate and revocation schemes for anonymous communications. The drawback of this solution is the high computational cost to compute pairings.

A game-theoretic approach to anonymous networking in the context of wireless networks is proposed by Venkitasubramaniam *et al.* in [28]; anonymity is quantified by the conditional entropy of the routes, and specific network design strategies are proposed to balance throughput and route anonymity, which is achieved by combining packet relay and injection of dummy traffic. Our proposed pseudonymization infrastructure also relies on packet forwarding, but it does not adopt dummy traffic injection.

A pseudonym-based infrastructure based on one-way hash functions is adopted by Henrici *et al.* [29] in the context of radio-frequency identification systems. The main idea is to use pseudonyms that change regularly and are linked to the owner of a tag, without affecting location privacy. The pseudonyms are computed collecting inputs from each node on the path to the receiver. The main disadvantage of the infrastructure is that it is static and thus cannot ensure long term security.

Burkhart *et al.* [30] propose security through private information aggregation, a library that allows efficient aggregation of multi-domain network data and preserves privacy using multiparty computation (MPC). Their proposal includes efficient MPC comparison operations and MPC protocols for events correlation and distinct counts computation. Moreover, they implement the protocols in security through private information aggregation library, evaluating the performance on realistic settings.

Ozdemir *et al.* [31] propose a data aggregation and authentication protocol for wireless sensor networks, which supports false data injection by a fraction of compromised nodes by verifing integrity directly on the encrypted data. However, these two papers present a solution for aggregating data, but they do not consider pseudonymization, which is the aim of the security infrastructure proposed in this paper.

The previously proposed methods are not suited for an anonymized Smart Grid deployment, because they do not satisfy the condition of low computational cost at the metre. Our approach solves this problem by using an SSS scheme. Moreover, we jointly address both the problem of frequent re-pseudonymization and of IDs recovery, by leveraging on a single pseudonymization protocol. The former is addressed by a multiple tier

pseudonymization network, while the latter exploits a novel key escrow procedure.

3. BACKGROUND

This section provides a short overview of the cryptographic schemes used in the pseudonymization protocol.

3.1. Shamir secret sharing scheme

Shamir secret sharing [32] is a threshold scheme proposed to divide a secret in w parts called *shares*. The shares are distributed among the participants to the protocol; in order to recover the secret, at least $t \leqslant w$ participants must cooperate.

The scheme works as follows: let $\mu \in Z_q$ be the secret, where q is a prime number, greater than all the possible secrets. To split the secret in w shares, generate $t-1$ integer random numbers $\rho_1, \rho_2, \cdots, \rho_{t-1}$ uniformly distributed in $[0, q-1]$ and compute the s-th share (x_s, y_s), for $1 \leqslant s \leqslant w$, where x_s are distinct integer numbers and $y_s = \mu + \rho_1 x_s + \rho_2 x_s^2 + \ldots + \rho_{t-1} x_s^{t-1} \mod q$. The secret can be reconstructed if t or more shares are available by using the Lagrange interpolation method.

Moreover, SSS is known to be a *perfect* secret sharing scheme [33]. Therefore, for any subset $\mathcal{S}$ of shares of cardinality at most $t-1$, it holds that

$$\Pr(M = \mu | \mathcal{S}) = Pr(M = \mu)$$

for every $\mu \in Z_q$, where M is the random variable indicating the secret chosen by the dealer.

3.2. Chaum mixing

Chaum [34] presents a technique based on public key cryptography that permits one correspondent to remain anonymous to a second, while allowing the second to respond via an untraceable return address. This technique relies on a *mixer* that processes each message before it is delivered. The use of mixing guarantees anonymity by hiding the correspondence between the sender and the receiver, which is achieved by wrapping the messages with a public-key cryptography. The Chaum mixing scheme is deployable, usable and has a simple design and that is why is widespread in numerous scenarios, [13–16].

The algorithm works as follows: a participant prepares a message $\mathcal{M}$ for delivery to a receiver at address A by sealing it with the addressee's public key K_a, appending the address A, and then sealing the results with the mixer's public key K_1. The mixer receives the encrypted message $K_1(R_1, K_a(R_0, \mathcal{M}), A)$, where R_0 and R_1 are random strings. Then, it decrypts the input with its private key, removes R_1, and outputs $K_a(R_0, \mathcal{M}), A$. Finally, the addressee decrypts the message with its private key, removes R_0 and obtains the original message $\mathcal{M}$.

3.3. Identity based proxy re-encryption

Green and Ateniese [35] propose an IB-PRE protocol based on the assumed intractability of the decisional bilinear Diffie–Hellman problem (DBDH) [35, Definition 3.2] in $\mathbb{G}_1, \mathbb{G}_T$. The IB-PRE scheme allows a proxy to convert an encryption under a user's identity into an encryption computed under another user's identity. Moreover, the proxy does not learn the secret keys of the users nor the plaintext. Furthermore, this scheme guarantees unidirectionality, meaning that a user A can delegate to another user B, without permitting A to decrypt B's ciphertexts, and non-interactivity, meaning that a user A can construct a re-encryption key without the participation of B. The IB-PRE has been adopted to secure communications and provide anonymity in many different frameworks [17–23].

It comprises the following set of algorithms:

- `Setup`(1^l) accepts a security parameter, l, and outputs both the master public parameters, $params$, which are distributed to users, and the master secret key, msk, which is kept private. Let $e : \mathbb{G}_1 \times \mathbb{G}_2 \rightarrow \mathbb{G}_T$ be a bilinear map, where $\mathbb{G}_1 = \langle g \rangle$ and $\mathbb{G}_T$ have order q. Let $\mathcal{H}_1, \mathcal{H}_2$ be independent full-domain hash functions $\mathcal{H}_1 : \{0,1\}^* \rightarrow \mathbb{G}_1$ and $\mathcal{H}_2 : \mathbb{G}_T \rightarrow \mathbb{G}_1$. To generate the scheme parameters, selects $s \leftarrow \mathbb{Z}_q^*$, and outputs $params = (\mathbb{G}_1, \mathcal{H}_1, \mathcal{H}_2, g, g^s), msk = s$.
- `KeyGen`$(params, msk, id)$ on input an identity, id, and the master secret key, msk, outputs a decryption key, sk_{id} corresponding to that identity. To extract a decryption key for identity $id \in \{0,1\}^*$, returns $sk_{id} = \mathcal{H}_1(id)^s$.
- `Encrypt`$(params, id, m)$ on input a set of public parameters, an identity id and a plaintext, $m \in \mathcal{M}$, where $\mathcal{M}$ is the messages space, outputs c_{id}, the encryption of m under the specified identity. To encrypt m, select $r \leftarrow \mathbb{Z}_q^*$ and output $c_{id} = (g^r, m \cdot e(g^s, \mathcal{H}_1(id))^r) = \langle C_1, C_2 \rangle$.
- `RKGen`$(params, sk_{id_1}, id_1, id_2)$ on input a secret key sk_{id_1} and identities id_1, id_2, outputs a re-encryption key, $rk_{id_1 \rightarrow id_2}$. It selects $X \leftarrow \mathbb{G}_T$ and computes $\langle R_1, R_2 \rangle = Encrypt(params, id_2, X)$ and returns $rk_{id_1 \rightarrow id_2} = \left\langle R_1, R_2, sk_{id_1}^{-1} \cdot \mathcal{H}_2(X) \right\rangle = \langle R_1, R_2, R_3 \rangle$.
- `Reencrypt`$(params, rk_{id_1 \rightarrow id_2}, c_{id_1})$ on input a ciphertext c_{id_1} under identity id_1, and a re-encryption key, $rk_{id_1 \rightarrow id_2}$, outputs a re-encrypted ciphertext c_{id_2}.
- `Decrypt`$(params, sk_{id}, c_{id})$ decrypts the ciphertext, c_{id} using the secret key sk_{id}, and outputs m or an `error`.

3.4. RSA cryptosystem with optimal asymmetric encryption padding

The Ron Rivest, Adi Shamir and Leonard Adleman (RSA) algorithm cryptosystem with optimal asymmetric encryption padding (OAEP) [33, Cryptosystem 5.4] is defined as

follows. Let $k_e = (b, e)$ and $k_d = (b, d)$ be the RSA public/private key pair with modulus b, which is l bits long, and encryption and decryption exponents, respectively, e and d. Let o be a positive integer with $o < l < 2o$. The deterministic one-way functions

$$H_1 : \{0,1\}^{l-o-1} \rightarrow \{0,1\}^o$$
$$H_2 : \{0,1\}^o \rightarrow \{0,1\}^{l-o-1}$$

are systemwide masking generation functions, which can be implemented using the construction in public-key cryptography standards #1 [36, Appendix B2].

The encryption function is defined as

$$E_{k_e} : \{0,1\}^o \times \{0,1\}^{l-o-1} \rightarrow \{0,1\}^l$$

The ciphertext $y = E_{k_e}(x, r)$ is calculated as follows:

$$x_1 = x \oplus H_1(r)$$
$$x_2 = r \oplus H_2(x_1)$$
$$E_{k_e}(x, r) = (x_1 \| x_2)^e \bmod b$$

The decryption function performs the following calculations:

$$x_1 = \mathcal{L}_{o+1}(y^d \bmod b)$$
$$x_2 = \mathcal{R}_{l-o-1}(y^d \bmod b)$$
$$x = H_1(x_2 \oplus H_2(x_1)) \oplus x_1$$

where $\mathcal{L}_n(x)$ and $\mathcal{R}_n(x)$ denote the n leftmost bits of x and the n rightmost bits of x, respectively.

Note that we use some different bit lengths with respect to the reference in order to guarantee that the modulus operation does not exceed the length limit. Moreover, note that we consider the RSA cryptosystem with OAEP secure against chosen-ciphertext attacks (CCA), as stated in [37].

3.5. Security against chosen-ciphertext attacks

In a CCA, the adversary has the ability not only to encrypt messages of her choice but also to request decryption of arbitrary ciphertexts. In fact, the adversary can access a decryption oracle $Dec_{sk}(\cdot)$ in addition to the encryption oracle $Enc_{pk}(\cdot)$. The only restriction to the oracle access is that the adversary is not allowed to request the decryption of the challenge ciphertext. A cryptosystem is assumed to be secure under a CCA if the adversary is not able to distinguish between the encryption of two arbitrary messages. The detailed description of the CCA indistinguishability experiment is given in [38].

Here, we report the description of the experiment $PubK^{cca}_{\mathcal{B},\Pi}(n)$ for a public key encryption scheme Π and an adversary $\mathcal{B}$:

(1) `Gen` (1^n) is run to obtain the keys (pk, sk).
(2) The adversary $\mathcal{B}$ is given pk and access to a decryption oracle $Dec_{sk}(\cdot)$. It outputs a pair of messages x_0, x_1 of the same length.
(3) A random bit $b \leftarrow \{0,1\}$ is chosen, and then a ciphertext $c \leftarrow Enc_{pk}(x_b)$ is computed and given to $\mathcal{B}$.
(4) $\mathcal{B}$ continues to interact with the decryption oracle but may not request a decryption of c itself. Finally, $\mathcal{B}$ outputs a bit b'.
(5) The output of the experiment is defined to be 1 if $b' = b$, and 0 otherwise.

It holds that

$$Pr\left[PubK^{cca}_{\mathcal{B},\Pi}(n) = 1\right] \leq \frac{1}{2} + negl(n)$$

where $negl(n) = \frac{1}{p(n)}$, for an arbitrary polynomial p, and a large integer n.

4. AN ARCHITECTURE FOR METERING DATA PSEUDONYMIZATION

4.1. The pseudonymization architecture

As depicted in Figure 1, three different sets of nodes are comprised in our proposed architecture:

- The set of smart *metres*, M, which generates the energy consumption data.
- The set of *privacy preserving nodes*, N, which are the nodes that perform data pseudonymization.
- The set of information *external entities*, E, which receives pseudonymized data and represents the utilities or other third party services.

The architecture also includes a *Configurator* node, which checks whether the monitoring requests received from the external entities are compliant to the grid privacy policies, periodically updates the public/private key pairs and recovers the metre's identities from their pseudonyms in case of emergencies or faults.

In the remainder of the paper, we assume that the communication network is reliable and timely, that is, no message can be lost because of communication delays or node malfunctioning. For an extensive fault-tolerance analysis of our privacy-preserving infrastructure, the reader is referred to [5]. Moreover, it is worth noting that our protocol is agnostic to the type of data to be anonymized. Therefore, our pseudonymization protocol can easily be integrated with data perturbation and obfuscation techniques; for example, as proposed in [39], a battery can be installed at the customer's premises in order to partially hide the energy consumption profile, thus reducing the accuracy of linking attacks based on behavioural patterns.

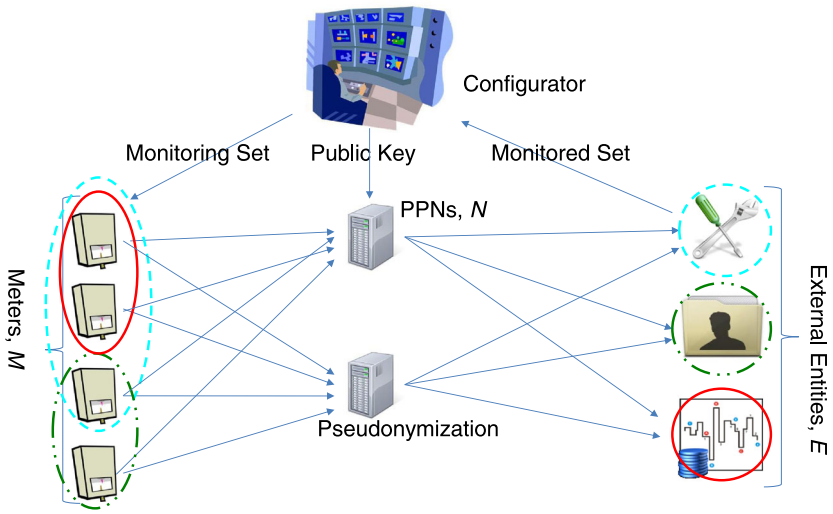

Figure 1. The pseudonymization architecture.

4.2. Problem statement

We assume that time is divided in intervals of given duration τ (in the order of seconds or minutes), and that all the nodes can be loosely synchronised to a common time reference. Each metre, PPN and External Entity is characterised by a unique identifier.

At each time interval, i, the m-th metre generates a measurement x_i^m, which is expressed as an integer number modulo q. During a setup phase, the e-th External Entity specifies the set of metres Π_e he/she wants to monitor. At every time interval, for each of the monitored metres, the External Entity expects to learn a set Ω_i^e of cardinality $|\Pi_e|$ of pseudonymized measurements:

$$\Omega_i^e = \left\{\left(x_i^m, PD_e^m\right) : m \in \Pi_e\right\} \tag{1}$$

where PD_e^m is the pseudonym of the m-th metre towards the e-th External Entity.

4.3. Scheme description

Our data pseudonymization protocol consists of the following tuple of probabilistic polynomial-time (p.p.t.) and polynomial time (p.t.) algorithms:

- $(k_d, params) \leftarrow \texttt{Setup}\left(1^l\right)$: takes as input the security parameter l and outputs the public parameters *params* and the Configurator's private key k_d.
- $\left(z_i^m(1), \ldots, z_i^m(n), \ldots, z_i^m(N), ID_m, r_i^m\right) \leftarrow \texttt{mSend}\left(param, i, m, x_i^m\right)$: during each round i, each metre m calls the mSend algorithm to encode its data x_i^m and then it sends the message msg_n^m, composed by the encrypted data $z_i^m(n)$, its identity ID_m and a nonce r_i^m, to the n-th PPN.
- $\left(PD_e^m, z_i^m(n)\right) \leftarrow \texttt{PPNSend}\left(param, i, n, ID_m, r_i^m, z_i^m(n)\right)$: at each time interval i, each PPN n encodes the metre's identity ID_m and sends the message $pmsg_n^m$, composed by the encrypted data $z_i^m(n)$ and the pseudonym PD_e^m, to an External Entity.
- $\left(PD_e^m, x_i^m\right) \leftarrow \texttt{eReceive}\left(param, i, e, PD_e^m, z_i^m(1), \ldots, z_i^m(n), \ldots, z_i^m(N)\right)$: finally, the External Entity e decodes the encrypted data and obtains the measurement x_i^m with the associated pseudonym PD_e^m.

We assume that the secret sharing scheme used in the algorithm mSend is unconditionally secure. Thus, the adversary is allowed to interact with an encryption oracle that encrypts a plaintext message x using the SSS scheme with threshold t and returning a ciphertext $z(t-1) \leftarrow Enc_t(x)$, where $z(t-1)$ is a vector of $t-1$ shares. A cryptosystem is unconditionally secure if the adversary is not able to distinguish the encryption of two arbitrary messages with less than t shares.
Now we describe the experiment $UnSec_{\mathcal{B},\Pi}$ for an encryption scheme Π and an adversary $\mathcal{B}$:

(1) A threshold t is chosen.
(2) The adversary $\mathcal{B}$ is given access to the encryption oracle $Enc_t(\cdot)$. It outputs a pair of messages x_0, x_1 of the same length.
(3) A random bit $b \leftarrow \{0,1\}$ is chosen, and then a ciphertext $z_b(t-1) \leftarrow Enc_t(x_b)$ is computed and given to $\mathcal{B}$. We call $z_b(t-1)$ the challenge ciphertext.
(4) $\mathcal{B}$ continues to interact with the encryption oracle. Finally, $\mathcal{B}$ outputs a bit b'.
(5) The output of the experiment is defined to be 1 if $b' = b$, and 0 otherwise.

It holds that

$$Pr(UnSec_{\mathcal{B},\Pi} = 1) = \frac{1}{2}$$

4.4. Security properties

In this section, we enlist the security properties that capture the privacy requirements of the Smart Grid infrastructure.

4.4.1. Full pseudonymization.

Consider the following experiment `full-p` for a given algorithm $\mathcal{A}$ and a parameter l; the experiment assumes an adversary as a malicious External Entity e^* and focuses on two metres $ID_1, ID_2 \in \Pi_{e^*}$.

(1) The `Setup`(1^l) algorithm outputs the system parameters.
(2) The first metre executes `mSend`$\left(param, i, 1, x_i^1\right)$ and outputs the messages $msg_1^1, \ldots, msg_n^1, \ldots, msg_N^1$.
(3) The second metre executes `mSend`$\left(param, i, 2, x_i^2\right)$ and outputs the messages $msg_1^2, \ldots, msg_n^2, \ldots, msg_N^2$.
(4) Each PPN n receives the two messages msg_n^1, msg_n^2 and calls the `PPNSend`$\left(param, i, n, ID_m, r_i^m, z_i^m(n)\right)$ algorithm. Then each PPN sends two messages $pmsg_n^m$ (with $m \in \{1, 2\}$) to the external entities.
(5) Finally, each External Entity runs `eReceive` $(param, i, e, PD_e^m, z_i^m(1), \ldots, z_i^m(n), \ldots, z_i^m(N)$ (with $m \in \{1, 2\}$) and obtains the measurement with the associated pseudonym.
(6) The malicious External Entity e^* executes $\mathcal{A}$ and outputs $m' \in \{1, 2\}$.
(7) The output of the experiment is 1 if $m' = m$, and 0 otherwise.

Definition 1. *A pseudonymization protocol provides full pseudonymization relatively to* `full-p` *if for all p.p.t. algorithms $\mathcal{A}$ there exists a negligible function negl such that*

$$Pr(\texttt{full-p} = 1) \leqslant \frac{1}{2} + negl(l)$$

4.4.2. Perfect forward anonymity

Consider the following modification to the `full-p` experiment for a given algorithm $\mathcal{A}$ and a parameter l, which we name `full-p-pfa` experiment. This assumes the presence of a malicious PPN n^* and a malicious External Entity e^* and focuses on two metres $ID_1, ID_2 \in \Pi_{e^*}$.

The `full-p` experiment is repeated until step 5 for some rounds $1, 2, \ldots, i$, thus, each round, the algorithms executed are `Setup`(1^l), `mSend`$\left(param, i, m, x_i^m\right)$, `PPNSend`$\left(param, i, n, ID_m, r_i^m, z_i^m(n)\right)$, and `eReceive` $\left(param, i, e, z_i^m(1), \ldots, z_i^m(n), \ldots, z_i^m(N)\right)$, all of them with $m \in \{1, 2\}$. Moreover, after the execution of step 5 and before step 6, during the round $i^*: i^* > i + \alpha\tau$, a collusion of a malicious External Entity e^* and a PPN n^* occurs. Such pair of malicious nodes can obtain the correspondence between the measurement $x_{i^*}^m$, the pseudonym $PD_{e^*}^m$, and the identity ID_m associated to a metre $m \in \{1, 2\}$. This happens because the malicious PPN n^* knows the correspondence between ID_m and $PD_{e^*}^m$, while the malicious External Entity e^* knows the correspondence between $PD_{e^*}^m$ and $x_{i^*}^m$. Then, the collusion executes the algorithm $\mathcal{A}$ and outputs $m' \in \{1, 2\}$. The output of the experiment is 1 if $m' = m$, and 0 otherwise.

Definition 2. *A pseudonymization protocol provides full pseudonymization with perfect forward anonymity relative to* `full-p-pfa` *if for all p.p.t. algorithms $\mathcal{A}$ there exist a negligible function negl such that*

$$Pr(\texttt{full-p-pfa} = 1) \leqslant \frac{1}{2} + negl(l)$$

4.4.3. Unconditionally indistinguishable encryption.

We define the following experiment `blind` for an adversary that controls a collusion of $t^* < t$ PPNs.

(1) The `Setup`(1^l) algorithm outputs the system parameters.
(2) At round i, the adversary chooses two secrets $\overline{x}_i^0$ and $\overline{x}_i^1$ and gives them to the metres.
(3) A random bit $b \in \{0, 1\}$ is chosen and kept secret to the adversary.
(4) The first metre executes `mSend`$\left(param, i, 1, \overline{x}_i^b\right)$ and outputs t messages msg_n^1 with the encrypted data $z_i^1(n)$, each of them being the share destined to the n-th PPN ($1 \leqslant n \leqslant t$).
(5) The second metre executes `mSend`$\left(param, i, 2, \overline{x}_i^{1-b}\right)$ and outputs t messages msg_n^2 with the encrypted data $z_i^2(n)$, each of them being the share destined to the n-th PPN.
(6) Each PPN n receives the two messages msg_n^1 and msg_n^2. The adversary outputs b'.
(7) The output of the experiment is 1 if $b' = b$, and 0 otherwise.

Definition 3. *A protocol provides unconditionally indistinguishable encryption under* `blind` *if it holds that*

$$Pr(\texttt{blind} = 1) = \frac{1}{2}$$

In Section 7.2, we provide the description of other properties related to our pseudonymization protocol.

5. THE PSEUDONYMIZATION FUNCTION

Let $E_{k_e}(\mu, r)$ be a keyed trapdoor one-way function. The function takes as input a plaintext μ and a security nonce r. The output of the function is the ciphertext y.

We assume that the Configurator generates the public/private key pair, keeps the private key k_d and distributes the public key k_e to all the PPNs. The cryptosystem allows the PPN $n \in N$ to compute the pseudonym PD_e^m, which will be associated to the data generated by metre $m \in M$ and destined to External Entity $e \in E$. The PPN calculates

$$PD_e^m = E_{k_e}\left(ID_m \| e \| \lceil i/\alpha \rceil \alpha, w_m^e\right) \tag{2}$$

The ciphering function E_{k_e} takes as input a concatenation of the metre's identity, ID_m, the External Entity ID number e, the round identifier i, and a security nonce w_m^e. As it will be detailed in Section 6, the frequent refreshment of w_m^e guarantees a prevention against linking attacks, as described in [12].

Note that such cryptosystem allows the Configurator to recover the metre identity by decrypting PD_e^m with its private decryption key k_d. In this paper, we consider RSA-OAEP as a randomised trapdoor function, because it is invertible, secure against CCA and it is one of the most widely used because of the easiness of implementation [40–43].

6. COMMUNICATION PROTOCOL

In this section, we describe the messages that constitute our proposed protocol, which exploits the homomorphic properties of SSS scheme to provide network blindness (property 3 in Section 4.2). Then, we discuss other two possible ways to provide the same property using, respectively, Chaum mixing and IB-PRE. In Section 8, we compare their performance, concluding that the Shamir-based one is more scalable. We stress that, while the mixing-based protocol is a straightforward implementation of [34], the IB-PRE-based one is an original elaboration over the ideas in [35]. In the initial version, however, the secret key and the re-encryption key were assumed to be held by the same entity. This is not the case with our protocol, therefore we need to prove that a node knowing the re-encryption key cannot recover the secret key. Such proof is provided in the Appendix.

All the protocols assume that a confidential, authenticated communication is established between the node pairs.

The data pseudonymization protocol consists of four phases:

(1) **Setup**: the initial phase is performed only once to define the set of public parameters and to distribute them to the users. Moreover, in this phase, each External Entity its set of monitored meters, the Configurator checks the admissibility of the aggregation requests received from the External Entities and communicates to each meter the set of External Entities which have included it in their monitoring set.
(2) **Key refresh**: this procedure is performed from time to time to update the key pairs and to communicate the new public keys to metres, PPNs and External Entities.
(3) **Data collection**: this phase is performed at every interval to collect the pseudonymized data and involves metres, External Entities, and PPNs.
(4) **Identity recovery**: this procedure is performed only in the presence of alarms/faults to recover the identity of the faulty metres and involves an External Entity and the Configurator.

We first describe the messages sent during the setup and the identity recovery, then we discuss the key refresh and data collection phases comparing the usage of SSS scheme to two alternative approaches relying on Chaum-mixing and IB-PRE.

During the initial setup phase, the following messages are exchanged:

(1.1) SPECIFYMONITOREDSET

$$e \rightarrow f: \Pi_e$$

The e-th External Entity specifies to the Configurator the set of metres, Π_e, that the External Entity wants to monitor. The Configurator checks the conformance of the External Entity's request to the system policy.

(1.2) SPECIFYMONITORINGSET

$$f \rightarrow m: \Psi_m$$

The Configurator computes the set Ψ_m of External Entities, which are monitoring metre m and communicates it to the metre.

In case of faults or alarms, an External Entity is allowed to obtain the identity of a metre (i.e. identity recovery) through the following steps:

(4.1) RECOVERYREQUEST

$$e \rightarrow f: PD_e^m$$

The e-th External Entity communicates to the Configurator, the pseudonym of the metre whose identity he is interested in. The Configurator deciphers PD_e^m using his private key k_d, removes $e \| \lceil \frac{i}{\alpha} \rceil \alpha$ and obtains ID_m.

(4.2) SENDIDENTITY

$$f \rightarrow e: PD_e^m \| ID_m$$

The Configurator communicates the metre's identity and the associated pseudonym to the External Entity.

6.1. Shamir secret sharing scheme

The SSS scheme works as follows. The measurements generated by every metre are divided in t shares, where t is a system parameter, and can be recovered if and only if all the shares are available at the External Entity (i.e. we assume $t = w$). We suppose that the number of installed PPNs is also equal to t. The metres send each share to a different PPN, therefore individual measurements can be obtained only through a collusion of all the involved PPNs. Once the n-th PPN receives a share from metre m destined to the External Entity e, it computes the metre's pseudonym, whose value depends both on m and e. Then, it forwards the share to the External Entity, together with the computed pseudonym (Figure 2). Therefore, the External Entity can recover the individual data by combining the shares associated to the same pseudonym but obtains no information about identity of the metres, which generated them.

With reference to Figure 3, the key refresh procedure includes only one message:

(2.1) REFRESH KEY

$$f \rightarrow n : k_e$$

The Configurator communicates to the PPNs, its public key k_e, every time the key pair (k_e, k_d) is refreshed. The key k_d is kept private.

During the data collection phase, the following messages are exchanged:

(3.1) SENDDATA

$$m \rightarrow n : s\left(x_i^m, n\right) \| ID_m \| r_i^m$$

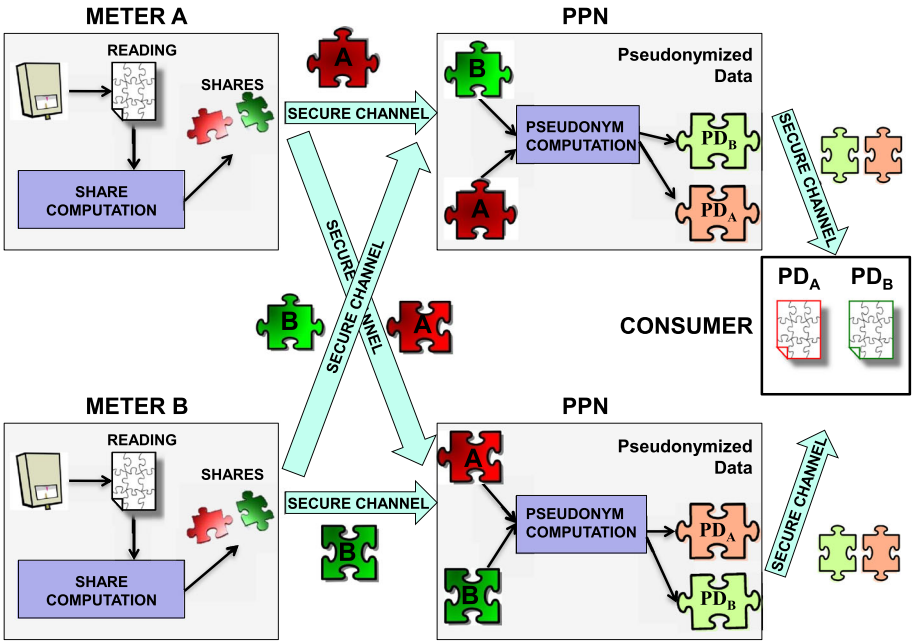

Figure 2. Shamir secret sharing scheme.

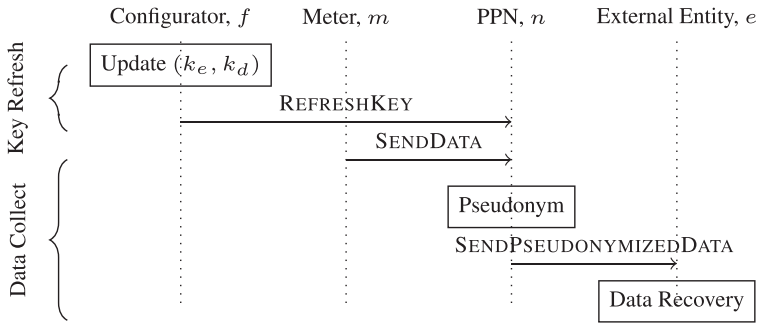

Figure 3. The Shamir secret sharing protocol.

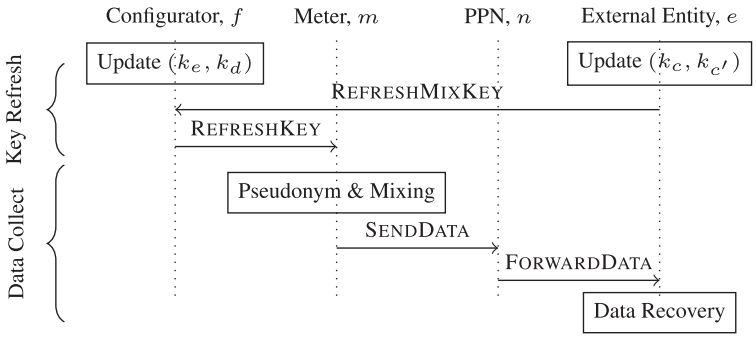

Figure 4. The mixing protocol.

At the i-th time interval, the metre m produces the measurement x_i^m (the secret) and sends to the n-th PPN the corresponding share $s\left(x_i^m, n\right)$ computed according to the SSS scheme, its identity ID_m and a random number r_i^m.

(3.2) SENDPSEUDONYMIZEDDATA

$$n \rightarrow e: s\left(x_i^m, n\right) \| PD_e^m$$

The n-th PPN computes the pseudonym PD_e^m according to Equation (2). The pseudonym will be associated to the data generated by m-th metre and destined to the e-th External Entity. To do so, the PPN uses the Configurator's public key k_e. Note that the security nonce w_m^e is updated with the current value of the hash-function $\mathcal{H}\left(r_i^m \| e\right)$, (which can be implemented using the construction in the public-key cryptography standards #1 [36, Appendix B2]), at all the i-th intervals such that i is an integer multiple of α, where α is a design parameter. Therefore, once w_m^e is refreshed, it remains unchanged for a time window of duration $T = \alpha\tau$, which represents the validity time span of the pseudonym.

Once the pseudonym is computed, the PPN sends it to the External Entity, together with the share. The External Entity waits until reception of all the t pseudonymized shares for each of the $|\Pi_e|$ pseudonyms and groups together the shares associated to the same pseudonym. Then, for each pseudonym it recovers the corresponding secret x_i^m.

6.2. Mixing approach

An alternative pseudonymization scheme relies on Chaum mixing; during the data collection phase, every metre generates the measurement x_i^m and computes the pseudonym PD_e^m. Then it creates the mixing packet $MIX_e^m = E_{k_c}\left[x_i^m \| PD_e^m\right]$, which includes both measurement and pseudonym, and sends it to a randomly chosen PPN through the SENDDATA message. The PPN forwards the packet (FORWARDDATA message) to the External Entity to whom the message is destined, which recovers the individual data by decrypting the packet. The key refresh phase is executed to update and refresh the key pairs $(k_c, k_{c'})$ for mixing and (k_e, k_d) for computing the pseudonyms.

Figure 4 shows the protocol messages of the key refresh and data collection phases.

6.3. Identity-based proxy re-encryption

A second variant of the pseudonymization protocol relies on the IB-PRE scheme. In this case, the key refresh phase comprises also the `KeyGen` algorithm that is executed by Configurator to generate the PPNs and External Entities' secret keys, sk_n and sk_e. The latter is sent with SENDSECRETKEY message. The Configurator also generates the re-encryption keys $rk_{n\rightarrow e}$ thanks to `RKGen` algorithm and sends them in the SENDREKEYING message to the each PPN. The keys are generated by the Configurator, because it is the only node that possesses the master secret key msk.

The data collection phase comprises the `Encrypt` algorithm performed by the metres to encrypt the measurements destined to the PPNs, the `Reencrypt` algorithm, and the computation of the pseudonyms performed by the PPNs. The messages SENDENCRYPTEDDATA and SENDREENCRYPTEDDATA are used to convey the encrypted data to the External Entities and are composed by the concatenation of the encrypted measurement y_n, the metre identity ID_m and a random number r_i^m, and the concatenation of the re-encrypted message y_e and the pseudonym PD_e^m, respectively. Finally, the `Decrypt` algorithm is used by the External Entities to decrypt the ciphertexts.

Figure 5 depicts the protocol messages in each phase.

In order to provide network blindness, the PPN cannot recover the secret key from the re-encryption key. A proof is given in the Appendix.

7. SECURITY EVALUATION

7.1. Security proofs

This section discusses how the properties presented in Section 4.2 are satisfied by our proposed pseudonymization cryptosystem. We do not discuss further the attack scenario of a passive intruder trying to collect multiple

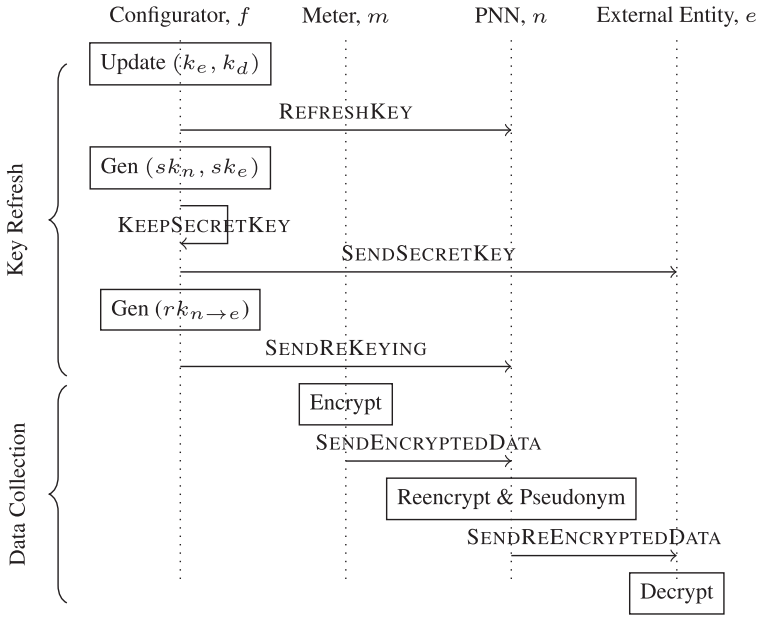

Figure 5. The proxy re-encryption protocol.

messages from a given metre to recover the individual measurements; the assumption of a computationally secure, confidential and authenticated channel between the nodes prevents this kind of attack. Moreover, we assume that the adversary $\mathcal{A}$ has no auxiliary information about the correspondence between the measurement x_i^m and the identity ID_m and thus cannot distinguish between two different measurements generated by different metres.

Theorem 1. *If the RSA with OAEP encryption scheme is CCA secure, then our pseudonymization protocol provides full pseudonymization with respect to* `full-p`.

Proof. By contradiction, let $\mathcal{A}$ be a p.p.t. algorithm that has more than a negligible advantage in the `full-p` experiment. Given the pseudonym PD_e^m and $(ID_m \| e \| \lceil i/\alpha \rceil \alpha)$, algorithm $\mathcal{A}$ yields 1 with non-negligible probability.
We now define the algorithm $\mathcal{B}$ that runs the CCA indistinguishability experiment where a challenge ciphertext $z = \overline{PD}_e^m$ is given to $\mathcal{B}$. Moreover, $\mathcal{B}$ chooses two plaintexts $(ID_m \| e \| \lceil i/\alpha \rceil \alpha)$ with $m \in \{1, 2\}$.
At point 4 of the CCA experiment, $\mathcal{B}$ interacts with $\mathcal{A}$, obtaining 1 if $\overline{PD}_e^m = E_{k_e}\left[ID_m \| e \| \lceil i/\alpha \rceil \alpha, r_i^m\right]$ with $m \in \{1, 2\}$, where E_{k_e} is defined in Section 3.4. The output of $\mathcal{A}$ is used as output of $\mathcal{B}$, solving the CCA experiment with non-negligible probability.
If $\mathcal{B}$ outputs 1, it means that $\mathcal{B}$ has solved the CCA indistinguishability experiment with non-negligible probability, that is, $Pr\left[PubK^{cca}_{\mathcal{B},\Pi}(n) = 1\right] \geqslant \frac{1}{2} + negl(n)$. □

Theorem 2. *If RSA with OAEP is CCA-secure, then our protocol provides full pseudonymization with perfect forward anonymity relative to* `full-p-pfa`.

Proof. By contradiction, let $\mathcal{A}$ be a p.p.t. algorithm that has more than a negligible advantage in the `full-p-pfa` experiment. Given the pseudonym PD_e^m and $(ID_m \| e \| \lceil i/\alpha \rceil \alpha)$, algorithm $\mathcal{A}$ yields the correct answer with probability greater than $1/2$. Moreover, $\mathcal{A}$ has an oracle access to a decryption function that gives the correspondence between $x_{i^*}^m$, PD_e^m and ID_m, relative to a time interval i^*. This means that $\mathcal{A}$ can say with certainty if $PD_e^m = E_{k_e}\left[ID_m \| e \| \lceil i^*/\alpha \rceil \alpha, r_{i^*}^m\right]$ is a valid relation. The output of $\mathcal{A}$ is used as the output of $\mathcal{B}$, defined in the previous proof, solving the CCA indistinguishability experiment with non-negligible probability, leading to the same proof of *Theorem 1*. □

Theorem 3. *If the SSS scheme is a perfect secret sharing scheme, then our protocol provides unconditionally indistinguishable encryption.*

Proof. Because the `blind` experiment assumes a collusion of $t^* < t$ PPNs, the colluded PPNs obtain two sets $\mathcal{S}_1, \mathcal{S}_2$, each of cardinality at most $t - 1$, of shares of the two secrets $\overline{x}_i^b$ and $\overline{x}_i^{1-b}$, respectively. Therefore

$$\begin{aligned} \Pr\{b = 0 | \mathcal{S}_1, \mathcal{S}_2\} &= \Pr\left\{M_1 = \overline{x}_i^0, M_2 = \overline{x}_i^1 | \mathcal{S}_1, \mathcal{S}_2\right\} \\ &= \Pr\left\{M_1 = \overline{x}_i^0 | \mathcal{S}_1, \mathcal{S}_2\right\} \end{aligned} \quad (3)$$

where M_1, M_2 are the random variables indicating the secrets encrypted by metre 1 and by metre 2, respectively.

Because the value of M_2 is completely determined by knowledge of M_1, then M_2 can be deleted from the last term of Equation (3).

Because the random polynomials used to generate $\mathcal{S}_1$ and $\mathcal{S}_2$ are independent, the knowledge of $\mathcal{S}_2$ gives no information about M_1. Further, exploiting the perfect secrecy property of SSS, we can write

$$\Pr\left\{M_1 = \overline{x}_i^0 | \mathcal{S}_1\right\} = \Pr\left\{M_1 = \overline{x}_i^0\right\} = \Pr\{b = 0\} = 1/2 \tag{4}$$

Similar considerations hold for $b = 1$. Therefore, the knowledge of $\mathcal{S}_1, \mathcal{S}_2$ gives no information about the value of b and no algorithm can guess b with probability greater than 1/2. □

7.2. Other security properties

(1) There exist a p.t. algorithm that, given the private key, can recover the identity of metre m from pseudonym PD_e^m.
This property is a direct consequence of the Configurator having the private key, which makes it able to recover ID_m from PD_m^e.
(2) Before sending its data, the metre is aware of the set of External Entities $\Psi_m = \{e : m \in \Pi_e\}$ monitoring its data thanks to the message SPECIFYMONITORINGSET .
(3) Given a pair of distinct metres' identities (m, m') and the same External Entity e, or a pair of distinct External Entities (e, e') and the same metre m, the output of the function E_{k_e} is always different. In other words, the output of the pseudonymization function is never the same for different sets of metres or External Entities, using the same value of e or m, respectively.
This property is a consequence of using the ciphering function E_{k_e} that relies on RSA with OAEP (Section 3.4), which guarantees that for different inputs, the outputs are never identical.

8. PERFORMANCE ASSESSMENT

In this section, we evaluate the computational costs of the protocol presented in Section 6 and the number of exchanged messages as a function of the system parameters $|M|$, $|N|$ and $|E|$. We also consider the case of a user, that is, a metre or an External Entity, joining or leaving the system.

First, it is useful to discuss a suitable choice for the system parameters: assuming 128-bit long identifiers for metres and External Entities, 64-bit long round numbers and 128-bit long nonces, a suitable choice is $o = 512$ and $l = 1024$, which results in 1024-bit pseudonyms. It is worth considering that, if the size of the pseudonym is an issue, the pseudonymization cryptosystem can be easily implemented using elliptic curve cryptography, resulting in shorter pseudonyms.

8.1. Number and size of exchanged messages

During the Setup and Identity Recovery phases the number of messages is independent from the choice of the measurement encryption scheme. In the setup phase, the Configurator receives $|E|$ messages from the External Entities and sends $|M|$ messages to the metres. For the identity recovery phase, the number of exchanged messages is at most $(2 \cdot |M| \cdot |E|)$, but assuming a low probability of faults, it tends to the lower bound, which is two messages (i.e. there is only one faulty metre).

Table I summarises the number of exchanged messages in the setup and identity recovery phases.

We consider now the exchanged messages during the key refresh and data collection phases.

During the Key Refresh phase, in case the SSS scheme is used, the Configurator simply forwards k_e to each of the $|N|$ PPNs. Conversely, in case of mixing scheme, each External Entity sends k_c to the Configurator, which in turn forwards the External Entities' public keys to the $|M|$ metres according to the monitoring requests. In the

Table I. Messages received and sent by the configurator, and the external entities during the setup and identity recovery phases.

	Configurator	
	No. of input messages	No. of output messages
Setup	$\|E\|$	$\|M\|$
IDRecovery	1 × *ID*s to be retrieved (worst case $\|E\| \times \|M\|$)	1 × *ID*s to be retrieved (worst case $\|E\| \times \|M\|$)
	External Entity	
	No. of input messages	No. of output messages
Setup	—	1
IDRecovery	1 × *ID*s to be retrieved (worst case $\|M\|$)	1 × *ID*s to be retrieved (worst case $\|M\|$)

DOI: 10.1002/ett

IB-PRE, the Configurator sends the messages containing the public keys and the re-encryption keys to the $|N|$ PPNs and sends the secret keys to the $|E|$ External Entities.

For what concerns the Data Collection phase, in the SSS scheme, the m-th metre sends a share to each of the $|N|$ PPN, which in turn sends the shares with the associated pseudonym to the External Entities that are monitoring the m-th metre. Therefore, the total number of exchanged messages is $|M| \times |N| + |M| \times |N| \times |E|$.

In the mixing scheme, the metre sends the $|E|$ mixing packets to the PPNs, which simply forward them to the External Entities. This procedure requires $2 \times |M| \times |E|$ messages.

Differently, in the IB-PRE scheme, the metre encrypts the measurement and sends it to only one PPN, which computes the pseudonym and re-encrypts the packet before forwarding it to the External Entities. In this scheme, the total amount of messages is $|M| + |M| \times |E|$.

We now evaluate the size of the messages. Let $L[x]$ be the length in bits of x. In the SSS scheme, the size of the SENDDATA message is $L\left[s\left(x_i^m, n\right)\right] + L[ID_m] + L\left[r_i^m\right] = 128+128+128 = 384$ bits, while the size of the SENDPSEUDONYMIZEDDATA message is $L\left[s\left(x_i^m, n\right)\right] + L\left[PD_e^m\right] = 128 + 1024 = 1152$ bits.

Table II. Comparison of the number of exchanged messages during the key refresh and data collection phases.

	Scheme	Input messages	Output messages	Output messages size [bit]
		Configurator		
KeyRefresh	Mixing	$\|E\|$	$\|M\|$	2048
	SSS	—	$\|N\|$	2048
	IB-PRE	—	$\|N\| + \|E\|$	4096+ 2048
		Metre		
Data Collect	Mixing	—	$\|E\|$	1152
	SSS	—	$\|N\|$	384
	IB-PRE	—	1	1504
		PPN		
Data Collect	Mixing	$\frac{\|M\|\times\|E\|}{\|N\|}$	$\frac{\|M\|\times\|E\|}{\|N\|}$	1024
	SSS	$\|M\|$	$\|M\| \times \|E\|$	1152
	IB-PRE	$\frac{\|M\|}{\|N\|}$	$\frac{\|M\|\times\|E\|}{\|N\|}$	3520
		External Entity		
KeyRefresh	Mixing	—	1	2048
	SSS	—	—	—
	IB-PRE	—	—	—
Data Collect	Mixing	$\|M\|$	—	—
	SSS	$\|M\| \times \|N\|$	—	—
	IB-PRE	$\|M\|$	—	—

SSS, Shamir secret sharing; IB-PRE, identity based proxy re-encryption.

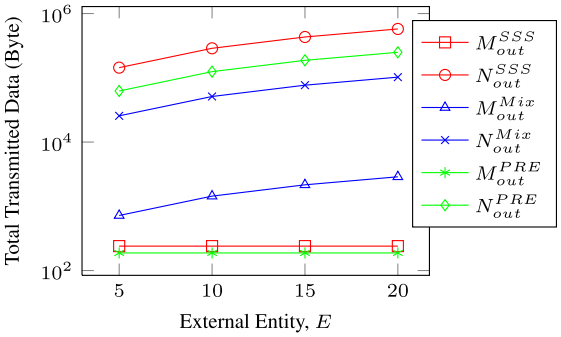

Figure 6. Comparison of the volume of the messages sent by each metre and PPN, assuming $|M| = 200$ and $|N| = 5$.

Conversely, in the mixing scheme, the metre sends the SENDDATA message to the PPN, which is $L[e] + L\left[MIX_e^m\right] = 128 + 1024 = 1152$ bits long, while the PPN sends only the 1024 bits long mixing packet $L\left[MIX_e^m\right]$ to the External Entity.

Finally, in the IB-PRE scheme, the metre sends the encrypted data to the PPN together with its identity and a round number, for a total length of $L[y_n] + L[ID_m] + L\left[r_i^m\right] = 1248 + 128 + 128 = 1504$ bits, while the PPN sends to the External Entity the re-encrypted message and the pseudonym, for a total message length of $L[y_e] + L\left[PD_e^m\right] = 2496 + 1024 = 3520$ bits. Therefore, the size of the single messages sent by each metre is lower in the SSS scheme than in mixing and IB-PRE schemes, while the size of the single messages sent by each PPN in the SSS scheme is slightly higher than in the mixing scheme, but lower than in the IB-PRE scheme.

Table II compares the number of messages received and sent by each entity and reports the corresponding message sizes.

Figure 6 depicts the trend of the total volume of input and output messages at the metres and PPNs, assuming $|M| = 200$ and $|N| = 5$, for different cardinalities of E. Because the metres are expected to stay in place for several years, it is important that the architecture is capable of scaling to larger numbers of metres and of External Entities without increasing the communication and computation cost at the metre. We note that, in the SSS scheme, the output data rate at the metres is only related to the number of PPNs, $|N|$, which is expected not to change over time. On the other hand, the number of metres and External Entities impacts onto the communication costs of the PPNs. However, PPNs are few with respect to the metres and easily upgradable in case of need.

8.2. Complexity and timing of cryptographic operations

In this section, we evaluate the computational complexity of the cryptographic operations in terms of asymptotic values and computational time. Because the setup and identity recovery phases are independent from the choice

of the measurement encryption scheme, we start with the evaluation of their computational costs.

Every time a user joins or leaves the system, the setup phase is re-executed and Π_e and Ψ_m are updated. In particular, if the new users are External Entities, they specify their Π_e to the Configurator, which checks the conformance of each request. Then the Configurator computes Ψ_m and communicates it to the metres. The same happens in case of new metres joining or leaving the system. Note that the costs of this phase are variable, and it is out of our scope to evaluate them. The same happens for the cost relative to the definition of the system parameters, which are omitted because it is performed only once.

The identity recovery phase involves an External Entity and the Configurator. The latter deciphers the pseudonym with his private key, exploiting the square and multiply algorithm, which has complexity $\mathbf{C}(RSA_{dec})$, which depends on the number of users to be de-anonymized $|M|$ requested by the $|E|$ External Entities.

During the Key Refresh phase, the Configurator chooses his public key k_e and computes the private key k_d, with complexity $\mathbf{C}(RSA_{gen})$. Conversely, the mixing scheme requires each External Entity to choose his public key k_c and to compute the corresponding private key $k_{c'}$ with complexity $\mathbf{C}(RSA_{gen})$. In the IB-PRE scheme, the Configurator performs the `KeyGen` and `RKGen` algorithms to generate the secret keys sk_n, sk_e, which remain unchanged, and the re-encryption key $rk_{n\rightarrow e}$, which is frequently changed. The computational costs are dominated by the `Weil Pairing` operations, which have complexity $\mathbf{C}(Pairing)$, which depends on p, which is a 1024 bits long prime number that corresponds to the field over which the elliptic curve is constructed.

The data collection phase is performed at every round i. In the SSS scheme, assuming $t = w = |N|$, the computation of the t shares requires the generation of $|N|-1$ integer random numbers, $|N|(|N|-1)$ modular multiplications and $|N|(|N|-1)$ modular sums. This operation has asymptotic complexity $\mathbf{C}(Share_{enc})$.

The PPNs have to compute the pseudonyms PD_e^m using cryptographically secure hash functions and RSA encryptions. The computational cost is dominated by the RSA encryption, which has complexity $\mathbf{C}(RSA_{enc})$. The External Entity receives all the shares associated to different pseudonyms and, for each pseudonym, recovers the corresponding secret with the Lagrange interpolation method, which has complexity $\mathbf{C}(Share_{join})$.

Differently, in the mixing scheme, the m-th metre computes the pseudonyms PD_e^m and creates the mixing packet MIX_e^m using cryptographically secure hash functions and RSA encryptions. The computational cost is dominated by the RSA encryption, which has complexity $2\cdot\mathbf{C}(RSA_{enc})$. The MIX_e^m message is sent to the PPNs that simply forwards the packet to the External Entity e whom the message is destined to. This operation has negligible complexity. The External Entity receives all the MIX packets and recovers the corresponding measurements performing the RSA decryption, which has complexity $\mathbf{C}(RSA_{dec})$.

Table III. Comparison of the computational costs ($\boldsymbol{C}$) during the setup, the identity recovery, the key refresh and the data collection (executed each round i) phases.

	Scheme	Complexity
		Configurator
Setup	All	*variable*
IDRecovery	All	$\boldsymbol{C}(RSA_{dec}) \times ID$s to be retrieved (worst case $\|E\| \times \|M\| \times \boldsymbol{C}(RSA_{dec})$)
KeyRefresh	Mixing	$\|M\|\boldsymbol{C}(RSA_{gen})$
	SSS	$\|N\|\boldsymbol{C}(RSA_{gen})$
	IB-PRE	$\|N\|\boldsymbol{C}(RSA_{gen}) + \|E\|\boldsymbol{C}(Pairing)$
		Metre
Data Collect *per* round	Mixing	$2\|E\|\boldsymbol{C}(RSA_{enc})$
	SSS	$\boldsymbol{C}(Share_{enc})$
	IB-PRE	$\boldsymbol{C}(Pairing)$
		PPN
Data Collect *per* round	Mixing	$\frac{\|M\|\times\|E\|}{\|N\|}$
	SSS	$\|M\| \times \|E\|\boldsymbol{C}(RSA_{enc})$
	IB-PRE	$\frac{\|M\|\times\|E\|}{\|N\|}(\boldsymbol{C}(RSA_{enc}) + \boldsymbol{C}(Pairing))$
		External Entity
KeyRefresh	Mixing	$\boldsymbol{C}(RSA_{gen})$
	SSS	—
	IB-PRE	—
Data Collect *per* round	Mixing	$\|M\|\boldsymbol{C}(RSA_{dec})$
	SSS	$\|M\|\boldsymbol{C}(Share_{join})$
	IB-PRE	$2\|M\|\boldsymbol{C}(Pairing)$

SSS, Shamir secret sharing; IB-PRE, identity based proxy re-encryption.

Finally, in the IB-PRE scheme, for the computation of the encrypted measurements, the metre has to perform the `HashToPoint` and `Weil Pairing` algorithms [44]. This operation has asymptotic complexity $\mathbf{C}(Pairing)$. The PPNs compute the pseudonyms PD_e^m using cryptographically secure hash functions and RSA encryptions and have to re-encrypt the measurements using the `Reencrypt` algorithm. The complexity is dominated by the RSA encryptions, which have complexity $\mathbf{C}(RSA_{enc})$, and by the encryption function, that has complexity $\mathbf{C}(Pairing)$. The External Entity receives all the encrypted measurements associated to different pseudonyms and recovers the corresponding secret by using the `Decrypt` algorithm, with complexity $\mathbf{C}(Pairing)$. The computational costs above discussed are summarized in Table III.

For the sake of completeness, in Table IV, we report the computational costs of the RSA, SSS and IB-PRE encryption and decryption procedures.

The computational time required by the implementation of IB-PRE scheme turns out to be much higher

Table IV. Timings of RSA keys generation, RSA encryption and decryption, share joining, re-encryption pairing and keys generation, assuming l=1024, t=5 and p=1024.

Operation	Timing
RSA_{gen}	7.23 s
RSA_{enc}	0.51 ms
RSA_{dec}	4.86 ms
$Share_{join}$	0.10 ms
Pairing	21.43 ms
KeyGen	98.69 ms
RKGen	43.24 ms

than in the mixing and SSS schemes. In fact, the Weil Pairing computation, which has the longer execution time, is repeated more than once per message and by every entity.

The previously discussed results show the following: (i) in the IB-PRE protocol, the number of exchanged messages is lower than in the mixing and SSS schemes, but the encryption time is longer; (ii) in the SSS scheme, the total number of exchanged messages is bigger than in the other two scenarios, but the execution time of the algorithm is shorter.

Hence, we can state that the SSS scheme provides the best compromise between number of messages and encryption time. In fact, although the total number of messages is high, their encryption is computed more quickly than the pairing of the IB-PRE scheme.

9. CONCLUSIONS

This paper proposes a pseudonymization protocol for smart metering measurements, in which the data gathered by the Smart Metres can be collected by multiple utilities and third parties without revealing the association between users' identities and pseudonyms. The pseudonymization procedure is performed at intermediate nodes called PPNs. We define the security properties that the protocol must satisfy and compare different implementations of the pseudonimyzation architecture, which leverage on the SSS scheme, on Chaum mixing and on an identity-based proxy re-encryption scheme, respectively. Results show that the Shamir-based protocol requires a processing effort that is suitable for real-time operations, even if it requires more bandwidth than the others.

APPENDIX A: SECURITY OF THE IB-PRE SCHEME

We prove that the PPN cannot recover the secret key in the IB-PRE scheme with security parameter l.

Theorem 4. *If the DBDH problem is intractable, then there not exists a p.p.t. algorithm $\mathcal{A}$ that, given the re-encryption key $rk_{id_1 \to id_2}$, can obtain the secret key sk_{id}.*

Proof. By contradiction, let $\mathcal{A}$ be a p.p.t. algorithm that has non-negligible probability $p(l)$ to obtain the secret key, given the re-encryption key. We use $\mathcal{A}$ to construct a second algorithm $\mathcal{B}$, which has non-negligible advantage in solving the DBDH problem. Algorithm $\mathcal{B}$ accepts as input a tuple $\langle \mathbb{G}_1 = \langle g \rangle, g^a, g^b, g^c, T \rangle$ and outputs 1 if $T = e(g,g)^{abc}$.

Having the re-encryption key $rk_{id_1 \to id_2}$ from algorithm $\mathcal{A}$, we know $\langle R_1, R_2, R_3 \rangle = \langle g^r, X \times e(g^s, \mathcal{H}_1(id_2))^r, sk_{id_1}^{-1} \times \mathcal{H}_2(X) \rangle$. Moreover, from $\mathcal{A}$, we obtain the correct $sk_{id_1} = \mathcal{H}_1(id)^s$ with non-negligible probability $p(l)$. Now, we assume as input for $\mathcal{B}$ the tuple $\langle \mathbb{G}_1 = \langle g \rangle, g^a = g^s, g^b = \mathcal{H}_1(id_2), g^c = g^r, T \rangle$, and as output 1 if $sk_{id_1}^{-1} = \mathcal{H}_2(R_2/T)$. If sk_{id} obtained from $\mathcal{A}$ is correct, then $\mathcal{B}$ gives the correct answer with probability 1. This happens with probability $p(l)$. If sk_{id} obtained from $\mathcal{A}$ is not correct, $\mathcal{B}$ gives a random answer, which is correct with probability 1/2. The overall probability that $\mathcal{B}$ gives the correct answer is $1/2 + p(l)/2$, which is larger than 1/2 by a non-negligible term, violating the assumption of intractability of the DBDH problem. □

Thus, we have proved that recovering the secret key from the re-encryption key is an intractable problem.

ACKNOWLEDGEMENT

Cristina Rottondi is funded by Fondazione Ugo Bordoni.

REFERENCES

1. National Institute of Standards and Technology (NIST). Guidelines for smart grid cyber security. *NIST Interagency Report 7628*, August 2010. http://www.nist.gov. [October 2012].
2. Committee on Homeland Security. Promoting and enhancing cybersecurity and information sharing effectiveness (PRECISE) act of 2011. *Bill H.R. 3674*, December 2011. http://www.gpo.gov/fdsys/pkg/BILLS-112hr3674ih/pdf/BILLS-112hr3674ih.pdf. [October 2012].
3. Cormode G, Srivastava D. Anonymized data: generation, models, usage, In *Proceedings of the 2009 ACM SIGMOD international Conference on Management of Data,* SIGMOD '09, ACM : New York, NY, USA, 2009; 1015–1018, DOI: 10.1145/1559845.1559968. http://doi.acm.org/10.1145/1559845.1559968. [October 2012].
4. Rottondi C, Mauri G, Verticale G. A data pseudonymization protocol for smart grids, In *IEEE Online Conference on Green Communications*, September 2012.
5. Rottondi C, Verticale G, Capone A. Privacy-preserving smart metering with multiple data consumers. *Computer Networks* 2013; **57**(7): 1699–1713. DOI: 10.1016/

j.comnet.2013.02.018. http://www.sciencedirect.com/science/article/pii/S1389128613000364. [October 2012].
6. Lee A, Zafar M. Energy data center. Briefing Paper, September 2012. http://www.cpuc.ca.gov/NR/rdonlyres/8B005D2C-9698-4F16-BB2B-D07E707DA676/0/EnergyDataCenterFinal.pdf. [October 2012].
7. Sui H, Wang H, Lu MS, jen Lee W. An ami system for the deregulated electricity markets. *IEEE Transactions on Industry Applications* 2009; **45**(6): 2104–2108. DOI: 10.1109/TIA.2009.2031848.
8. Kreutzmann H, Vollmer S, Tekampe N, Abromeit A. Protection profile for the gateway of a smart metering system (gateway PP), Federal Office for Information Security, Germany, August 2011.
9. Stegelmann M, Kesdogan D. Gridpriv: A smart metering architecture offering k-anonymity, In *2012 IEEE 11th International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom)*, Liverpool, United Kingdom, 2012; 419–426, DOI: 10.1109/TrustCom.2012.170.
10. Rusitschka S, Gerdes C, Eger K. A low-cost alternative to smart metering infrastructure based on peer-to-peer technologies, In *Energy Market, 2009. EEM 2009. 6th International Conference on the European*, 2009; 1–6, DOI: 10.1109/EEM.2009.5311431.
11. Cavoukian A, Polonetsky J, Wolf C. Smartprivacy for the smart grid: embedding privacy into the design of electricity conservation. In *Identity in the Information Society*. Springer: Netherlands, April 2010; 3(2); 275-294, DOI: 10.1007/s12394-010-0046-y.
12. Jawurek M, Johns M, Rieck K. Smart metering depseudonymization, In *Proceedings of the 27th Annual Computer Security Applications Conference,* ACSAC '11, ACM: New York, NY, USA, 2011; 227–236, DOI: 10.1145/2076732.2076764.
13. Rennhard M. Introducing morphmix: Peer-to-peer based anonymous internet usage with collusion detection, In *Proceedings of the Workshop on Privacy in the Electronic Society (WPES 2002*, Washington, DC, USA, 2002; 91–102.
14. Danezis G, Dingledine R, Hopwood D, Mathewson N. Mixminion: Design of a type III anonymous remailer protocol, In *Proceedings of the 2003 IEEE Symposium on Security and Privacy*, Oakland, California, USA, 2003; 2–15.
15. Berthold O, Federrath H, Kopsell S. Web MIXes: A system for anonymous and unobservable internet access, In *International Workshop on Designing Privacy Enhancing Technologies: Design Issues in Anonymity and Unobservability*, Springer-Verlag New York, Inc., 2001; 115–129.
16. Gomułkiewicz M, Klonowski M, Kutyłowski M. Rapid mixing and security of Chaum's visual electronic voting. In *Computer Security ESORICS 2003*, Vol. 2808, Snekkenes E, Gollmann D (eds), Lecture Notes in Computer Science. Springer: Berlin Heidelberg, 2003; 132–145, DOI: 10.1007/978-3-540-39650-5 8. http://dx.doi.org/10.1007/978-3-540-39650-5_8. [October 2012].
17. Clarke I, Sandberg O, Wiley B, Hong TW. Freenet: a distributed anonymous information storage and retrieval system, In *International Workshop on Designing Privacy Enhancing Technologies: Design Issues in Anonymity and Unobservability*, Springer-Verlag New York, Inc., 2001; 46–66.
18. Shao J. Anonymous id-based proxy re-encryption. In *Information Security and Privacy*, Vol. 7372, Susilo W, Mu Y, Seberry J (eds), Lecture Notes in Computer Science. Springer: Berlin Heidelberg, 2012; 364–375, DOI: 10.1007/978-3-642-31448-3_27. http://dx.doi.org/10.1007/978-3-642-31448-3_27. [October 2012].
19. Hur J, Yoon H. A multi-service group key management scheme for stateless receivers in wireless mesh networks. *Mobile Networks and Applications* 2010; **15**(5): 680–692. DOI: 10.1007/s11036-009-0191-4. http://dx.doi.org/10.1007/s11036-009-0191-4. [October 2012].
20. Suriadi S, Foo E, Smith J. Conditional privacy using re-encryption, In *Proceedings of the 2008 IFIP International Conference on Network and Parallel Computing,* NPC '08, IEEE Computer Society: Washington, DC, USA, 2008; 18–25, DOI: 10.1109/NPC.2008.37. http://dx.doi.org/10.1109/NPC.2008.37. [October 2012].
21. Yang Y, Zhang Y. A generic scheme for secure data sharing in cloud, In *Proceedings of the 2011 40th International Conference on Parallel Processing Workshops,* ICPPW '11, IEEE Computer Society: Washington, DC, USA, 2011; 145–153, DOI: 10.1109/ICPPW.2011.51. http://dx.doi.org/10.1109/ICPPW.2011.51. [October 2012].
22. Chim T, Yiu S, Hui L, Li V. VSPN: VANET-based secure and privacy-preserving navigation. *IEEE Transactions on Computers* 2012; **PP**(99): 1–1. DOI: 10.1109/TC.2012.188.
23. Sahu RA, Padhye S. Identity-based multi-proxy multi-signature scheme provably secure in random oracle model. *Transactions on Emerging Telecommunications Technologies* 2013. DOI: 10.1002/ett.2667. http://dx.doi.org/10.1002/ett.2667. [October 2012].
24. Efthymiou C, Kalogridis G. Smart grid privacy via anonymization of smart metering data, In *2010 First IEEE International Conference on Smart Grid Communications (SmartGridComm)*, Gaithersburg, Maryland, USA, October 2010; 238–243, DOI: 10.1109/SMARTGRID.2010.5622050.

25. Rial A, Danezis G. Privacy-preserving smart metering, In *Proceedings of the 10th Annual ACM Workshop on Privacy in the Electronic Society,* WPES '11, ACM: New York, NY, USA, 2011; 49–60, DOI: 10.1145/2046556.2046564.
26. Zhang Y, Liu W, Lou W. Anonymous communications in mobile ad hoc networks, INFOCOM 2005. 24th Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings IEEE, Miami, FL, USA, Vol. 3, March 2005; 1940–1951, DOI: 10.1109/INFCOM.2005.1498472.
27. Huang D. Pseudonym-based cryptography for anonymous communications in mobile ad hoc networks. *International Journal of Security and Networks* 2007; **2**(3/4): 272–283. Interscience Enterprises Ltd: Arizona, AZ, USA.
28. Venkitasubramaniam P, Tong L. A game-theoretic approach to anonymous networking. *IEEE/ACM Transactions on Networking* 2012; **20**(3): 892 –905. DOI: 10.1109/TNET.2011.2176511.
29. Henrici D, Gotze J, Muller P. A hash-based pseudonymization infrastructure for RFID systems, In *Second International Workshop on Security, Privacy and Trust in Pervasive and Ubiquitous Computing, 2006. SecPerU 2006*, Lyon, France, June 2006; 6–27.
30. Burkhart M, Strasser M, Many D, Dimitropoulos X. SEPIA: Privacy-preserving aggregation of multi-domain network events and statistics, In *Usenix Security Symposium,* USENIX, Washington DC, USA, 2010.
31. Ozdemir S, Cam H. Integration of false data detection with data aggregation and confidential transmission in wireless sensor networks. *IEEE/ACM Transactions on Networking* 2010; **18**(3): 736 –749. DOI: 10.1109/TNET.2009.2032910.
32. Shamir A. How to share a secret. *Communications of the ACM* 1979; **22**: 612–613.
33. Stinson D. *Cryptography Theory and Practice*, (2nd edn). CRC Press: Boca Raton, FL, USA, 2005.
34. Chaum DL. Untraceable electronic mail, return addresses, and digital pseudonyms. *Communications of the ACM* 1981; **24**(2): 84–90. DOI: 10.1145/358549.358563.
35. Green M, Ateniese G. Identity-based proxy re-encryption. In *Applied Cryptography and Network Security*, Vol. 4521, Katz J, Yung M (eds), Lecture Notes in Computer Science. Springer: Berlin / Heidelberg, 2007; 288–306, DOI: 10.1007/978-3-540-72738-5.
36. Jonsson J. KB. Public-key cryptography standards (PKCS) #1: RSA cryptography, specifications version 2.1, RSA Laboratories, 2003.
37. Shoup V. Oaep reconsidered. In *Advances in Cryptology – CRYPTO 2001*, Kilian J (ed.), Lecture Notes in Computer Science. Springer Berlin Heidelberg, 2001; 239–259, DOI: 10.1007/3-540-44647-8_15. http://dx.doi.org/10.1007/3-540-44647-8_15. [October 2012].
38. Katz J, Lindell Y. *Introduction to Modern Cryptography (Chapman & Hall/CRC Cryptography and Network Security Series)*. Chapman & Hall/CRC: Boca Raton, FL, USA, 2007.
39. Varodayan D, Khisti A. Smart meter privacy using a rechargeable battery: Minimizing the rate of information leakage, In *2011 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, May 2011; 1932 –1935, DOI: 10.1109/ICASSP.2011.5946886.
40. Bellare M, Boldyreva A, Desai A, Pointcheval D. Key-privacy in public-key encryption. In *Lecture notes in Computer Science*. Springer-Verlag: Heidelberg, Germany, 2001; 566–582.
41. Hayashi R, Okamoto T, Tanaka K. An RSA family of trap-door permutations with a common domain and its applications. In *Public Key Cryptography PKC 2004*, Vol. 2947, Bao F, Deng R, Zhou J (eds), Lecture Notes in Computer Science. Springer: Berlin Heidelberg, 2004; 291–304, DOI: 10.1007/978-3-540-24632-9_21. http://dx.doi.org/10.1007/978-3-540-24632-9_21. [October 2012].
42. Bellare M, Boldyreva A, O'Neill A. Deterministic and efficiently searchable encryption, In *Proceedings of the 27th Annual International Cryptology Conference on Advances in Cryptology,* CRYPTO'07, Springer-Verlag: Berlin, Heidelberg, 2007; 535–552. http://dl.acm.org/citation.cfm?id=1777777.1777820. [October 2012].
43. Malone-Lee J, Mao W. Two birds one stone: signcryption using RSA, In *Proceedings of the 2003 RSA Conference on the Cryptographers' Track,* CT-RSA'03, Springer-Verlag: Berlin, Heidelberg, 2003; 211–226. http://dl.acm.org/citation.cfm?id=1767011.1767032. [October 2012].
44. Boneh D, Franklin M. Identity-based encryption from the Weil pairing. In *Advances in Cryptology — CRYPTO 2001*, Vol. 2139, Kilian J (ed.), Lecture Notes in Computer Science. Springer: Berlin / Heidelberg, 2001; 213–229. 10.1007/3-540-44647-8.