A Data Pseudonymization Protocol for Smart Grids

Cristina Rottondi, Giulia Mauri, and Giacomo Verticale
Dipartimento di Elettronica e Informazione, Politecnico di Milano, Piazza Leonardo da Vinci, 32, Milano, Italy
rottondi@elet.polimi.it, vertical@elet.polimi.it

***Abstract*—Privacy and protection of user-related data is of paramount importance in Smart Grid scenarios: on one hand, information regarding customers' personal habits can be inferred by analysing metering data; on the other hand, detailed knowledge of consumption measurements is crucial for the timely management of energy distribution, provisioning, and forecasting.**

This paper proposes a pseudonymization protocol for data gathered by the meters: the protocol relies on a network infrastructure that involves a set of Privacy Preserving Nodes (PPNs). These nodes perform data pseudonymization without having access to the measurements, which are masked by means of a secret splitting scheme. Multiple entities such as utilities and third parties are allowed to collect pseudonymized data, which maintain their temporal sequentiality along a time window of finite duration, but cannot relate them to the identities of the users that generated the data or to the data generated by the same user in the preceding or following time windows. The paper also provides an evaluation of the security and of the performance of the protocol.

***Index Terms*—Data Privacy; Data Pseudonymization; Smart Grid; Automatic Metering Infrastructure**

I. Introduction

Consumption traces collected by Smart Meters are highly privacy-sensitive data. Fine-grained measurements improve the quality of the information available to enhance electricity provision, add value to services for customers, and improve billing. However, this can also result in a violation of people's privacy, since it has been widely shown that users' personal habits and customs can be inferred by analysing energy consumption data gathered by the meters. Therefore, NIST [1] mandates that, unless strictly necessary, metering data should be pseudonymized in order to prevent utilities and third parties from linking the collected information to the identity of the customers that generated them. In particular, vendors and societies conducting target marketing should not have access to user-related data but only to anonymized measurements, since energy load profiles could for example be used to track the usage of particular electrical appliances [2].

One possible solution to the problem is providing the utilities only with aggregated data. In [3], we propose a security privacy-preserving infrastructure for data spatial and/or temporal aggregation for multiple data Consumers (such as utilities, third parties, and service providers) according to their needs. Paper [3] introduces in the smart grid architecture a set of Privacy Preserving Nodes (PPNs), which perform aggregation directly on the ciphered data by exploiting the homomorphic properties of Shamir's secret sharing scheme.

Cristina Rottondi is funded by Fondazione Ugo Bordoni

This paper explores a different solution, describing a pseudonymization protocol that allows the data Consumers to obtain disaggregated data without being able to associate them with the identity of the data Producer (i.e., the meter) that generated them. The pseudonymization protocol proposed in this paper satisfies the following requirements:

1) The Consumers do not have access to the Producer's identities, which are replaced by pseudonyms. The validity of a pseudonym is time limited: after the expiration time, the pseudonym must be substituted. Therefore, the data generated by the same Producer cannot be temporally linked over a time window longer than the validity time span.
2) A Producer monitored by multiple Consumers is associated to a different pseudonym for each Consumer. Therefore, a collusion of Consumers cannot infer additional information by comparing the pseudonyms of the monitored Producers.
3) The function mapping the user's identity in a pseudonym must guarantee that two different identities cannot be associated to the same pseudonym during the same validity time span.
4) The Producers know which are the Consumers they are monitored by.
5) The Configurator is the only node that, under particular conditions (alarms, faults,...), can recover the Producer's identity using its pseudonym. Alternatively, only a collusion of nodes can recover the association between a Producer's pseudonym, identity and measurement.
6) A pair of identical measurements originates two encrypted values that are indistinguishable from a pair of masked values originated by two non-identical measurements.

The remainder of the paper is structured as follows: Section II provides an overall view about data pseudonymization and anonymization in various contexts of communication networks, focusing on the Smart Grid scenario. Section III describes the Smart Grid framework, while the pseudonymization protocol is described in Section IV. Section V discusses the security guarantees the protocol provides. An implementation of the protocol is described in Section VI, while the performance assessment of the protocol is provided in Section VII. Concluding remarks are left for the final Section.

II. Related Work

The problem of metering data pseudonymization in the context of Smart Grids has recently attracted the interest

978-1-4799-0396-2/12/$31.00 ©2012 IEEE

of numerous researchers. Efthimiou and Kalogridis in [4] describe a method for anonymization of electrical metering data sent by smart meters. They suppose to have two different IDs embedded in the smart meter: one, the High Frequency ID (HFID), is anonymous, while the other, the Low Frequency ID (LFID), is attributable. The meter manufacturer is the only one that knows the correspondence between HFID and LFID. This method provides an additional level of security thanks to the trust level of such an escrow service but it may not offer an adequate privacy protection. In fact some attacks, e.g. the linking attacks provided in [5], have access to anonymous consumption traces but still obtain information from secondary sources. Differently, our protocol relies on single IDs and does not require them to be hard-coded in the smart meter itself.

Jawurek et al. [5] 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 using patterns in electricity consumption. The authors also analyse three mitigation techniques: lower resolution, frequent re-pseudonymization and privacy preserving techniques. Our protocol also relies on frequent re-pseudomymization, but it does not perform data aggregation and can be applied also to fine-grained data.

A privacy preserving protocol is presented in [6]. In this scheme, the meter 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 but is used only for billing purposes, while our protocol is agnostic to the type of user-related data subjected to anonymization.

Privacy protection is an important topic also in other contexts, from of mobile ad-hoc networks (MANET) to RFID systems and health-care. Current anonymity research in wireless networks mainly focus on unlinkable and unobservable data-delivering, which consists in encrypting packet contents and hiding the header fields, or on anonymous routing protocols aimed at preventing the adversaries from learning the packet forwarding path.

Public-key based solution have proposed to guarantee communication anonymity, which means that sender's and receiver's identities are hidden to external observers. In particular, [7] proposes a novel pairing-based anonymous on-demand routing protocol. In this approach, a Trust Authority (TA) 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 (AU). The protocol in [7] guarantees sender anonymity, receiver anonymity and relationship anonymity, which means that an adversary is not able to determine the sender and the receiver of a packet, neither the two end points of the communication. However, the anonymous communications are not anonymous to the TA. To solve this problem, Huang in [8] proposes pairing-based encryption/decryption, key exchange, blind certificate and revocation solutions for anonymous communications.

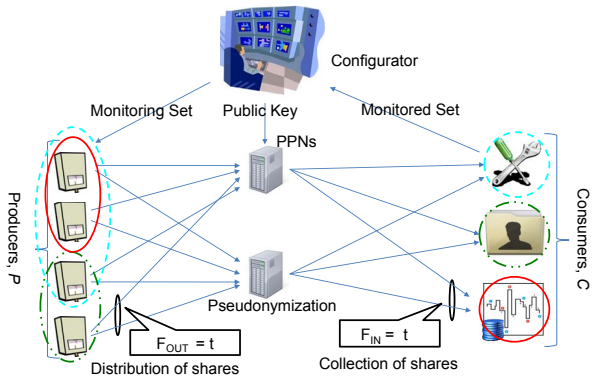

Fig. 1. Pseudonymization Architecture

A pseudonym-based infrastructure is adopted by Henrici et al. [9] in the context of RFID 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 main issue with that approach is that the infrastructure is static and also vulnerable to collecting node identifiers and abusing them.

Privacy protection is one of the fundamental issues also in health care, where a trade-off between the patient's needs for privacy and the society's needs to improve efficiency and reduce costs of the health care system is needed. Riedl et al. [10] present a new architecture for the pseudonymization of medical data, based on a layered structure with authorization mechanisms. The privacy is assured by securing the link between the patient's identification data and his/her anamnesis data with the encryption of the identification data with a pseudonymization key. Health care providers are allowed to decrypt the data only with the authorization of the patient. This system grants that the patients remain in full control of their data and can revoke a given authorization.

III. The Pseudonymization Architecture

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

- the set of information *Producers*, P, which represent the smart meters;
- the set of *Privacy Preserving Nodes* (PPN), N, which are the nodes that perform data pseudonymization;
- the set of information *Consumers*, C, which receive pseudonymized data and represent the utilities or other third party services.

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

The pseudoymization procedure relies on a secret splitting scheme: the measurements generated by every Producer 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 Consumer. As shown in Figure 1, the Producers send each share to a different PPN, therefore individual measurements can be obtained only through a collusion of all the involved PPNs.

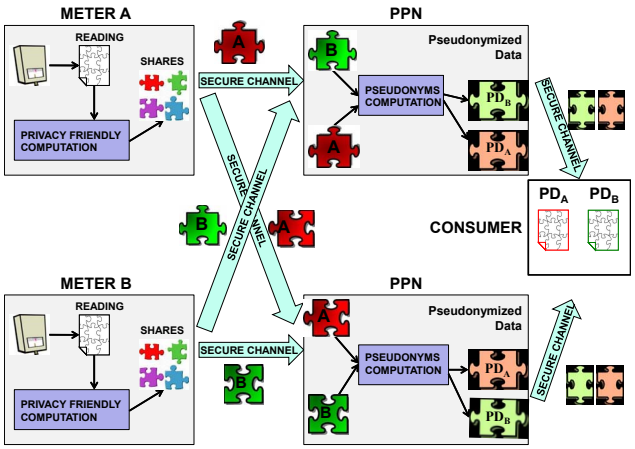

Fig. 2. Pseudonymization Protocol

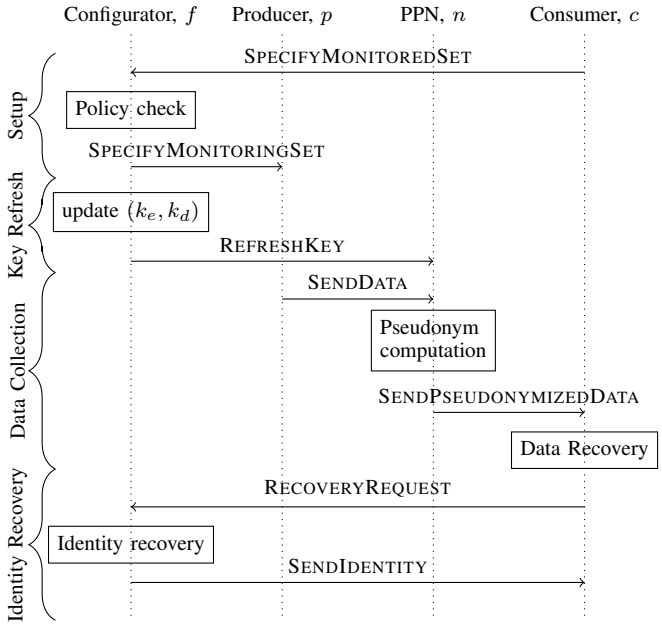

Fig. 3. The Pseudonymization Protocol

Once the PPN $n \in N$ receives a share from Producer $p \in P$ destined to Consumer $c \in C$, it computes the Producer's pseudonym, whose value depends both on p and c. Then, it forwards the share to the Consumer, together with the computed pseudonym (see Figure 2). Therefore, the Consumer can recover the individual data by combining the shares associated to the same pseudonym, but obtains no information about identity of the Producers who generated them.

IV. A Pseudonymization Protocol for Smart Grids

We assume that all Consumers and PPNs are identified by a unique number, that all the nodes are loosely synchronized to common time reference and that time is divided in rounds of duration τ. Supposing the usage of a secret splitting scheme with t shares, the number of installed PPNs is also equal to t. We also assume that the communication channel between Producers and PPNs is confidential and authenticated. We define the following cryptosystem that will be used in the anonymization protocol:

Pseudonymization Cryptosystem: Let E_{k_e} be a keyed trapdoor one-way function. The function takes as input a plaintext x and a security nonce r. The output of the function is the ciphertext y. The cryptosystem is characterized by the following properties:

1) The cryptosystem relies on a public key encryption scheme and assumes that the Configurator possesses a public key/private key pair (k_e, k_d), and that the public key is known to all the PPNs.
2) If $x = x'$ and $r \neq r'$, then $y \neq y'$, where $y = E_{k_e}(x, r)$ and $y' = E_{k_e}(x', r')$.
3) If $x \neq x'$ and $r = r'$, then $y \neq y'$.
4) For any given r, the cryptosystem $E_{k_e}(x, r)$ is semantically secure [11].

Note that the cryptosystem is agnostic with respect to the encryption technique: a possible implementation and the assessment of its performance are proposed in Sections VI and VII.

The pseudonymization protocol consists of four phases:

- **Setup**: the initial phase is performed only once. During setup each Consumer specifies the set of monitored Producers, the Configurator checks the admissibility of the Consumers' requests and communicates to each Producer the set of Consumers interested in monitoring its data.
- **Key Refresh**: this procedure is performed from time to time to substitute the Configurator's key pair and to communicate the new public key to the PPNs.
- **Data Collection**: this phase is performed at every round to collect the pseudonymized data and involves Producers, Consumers, and PPNs.
- **Identity Recovery**: this procedure is performed only in presence of alarms/faults to recover the Producer's identity and involves a Consumer and the Configurator.

Figure 3 shows the protocol messages. Let i be the round number. During the initial configuration phase the following messages are exchanged:

1. SPECIFYMONITOREDSET

$$c \rightarrow f : \Pi_c$$

The Consumer c specifies to the Configurator the set of Producers it wants to monitor, Π_c. The Configurator checks the conformance of the Consumer's request to the system policy.

2. SPECIFYMONITORINGSET

$$f \rightarrow p : \Psi_p$$

The Configurator computes the set of Consumers Ψ_p which are monitoring Producer p and communicates it to the Producer.

The key refresh procedure includes one message:

3. REFRESHKEY

$$f \rightarrow n\colon k_e$$

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:

4. SENDDATA

$$p \rightarrow n\colon s(x_i^p, n) \| ID_p \| r_i^p$$

The Producer p sends to nth-PPN the corresponding share $s(x_i^p, n)$ of the measurement x_i^p, its identity ID_p and a random number r_i^p. For the computation of the t shares $s(x_i^p, n)$ $(n = 1, 2,...,t)$ the following procedure is performed: at round i, the Producer p produces the measurement x_i^p (the secret) and generates $t-1$ integer random numbers $q_1, q_2,...,q_{t-1}$ uniformly distributed in the range $[0, Q-1]$, where Q is an integer number, larger than all the possible secrets. Then, the Producer computes the t shares as follows: $s(x_i^p, 1) = q_1$, $s(x_i^p, 2) = q_2$, ..., $s(x_i^p, t-1) = q_{t-1}$, $s(x_i^p, t) = x_i^p - q_1 - q_2 - \cdots - q_{t-1}$.

5. SENDPSEUDONYMIZEDDATA

$$n \rightarrow c\colon s(x_i^p, n) \| PD_c^p$$

where

$$PD_c^p = E_{k_e}[ID_p \| c \| \lceil i/\alpha \rceil \alpha, w_p].$$

The nth-PPN computes the pseudonym PD_c^p which will be associated to the data generated by Producer p and destined to Consumer c. To do so, the PPN uses the Configurator's public key k_e. The ciphering function E_{k_e} takes as input the Producer's identity, the Consumer identification number c, the round identifier i and a security nonce w_p. The latter is updated with the current r_i^p at all the ith-rounds such that i is an integer multiple of α, where α is a design parameter. Therefore, once w_p is refreshed, it remains unchanged for a time window of duration $T = \alpha\tau$, which represents the validity time span of the pseudonym. The refreshment of w_p guarantees a prevention against linking attacks, as described in [5]. For a detailed definition of the implementation of the function E_{k_e}, the reader is referred to Section VI.

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

In case of faults or alarms, a Consumer is allowed to obtain the identity of a Producer through the following steps:

6. RECOVERYREQUEST

$$c \rightarrow f\colon PD_c^p$$

The Consumer c communicates to the Configurator the pseudonym of the Producer whose identity it is interested in. The Configurator deciphers PD_c^p using the private key k_d and obtains ID_p.

7. SENDIDENTITY

$$f \rightarrow c\colon PD_c^p \| ID_p$$

The Configurator communicates the Producer's identity and the associated pseudonym to the Consumer.

V. SECURITY EVALUATION OF THE PROPOSED PROTOCOL

This section discusses how the security requirements 1–6 in Section I are satisfied by our proposed protocol. We assume an *honest-but-curious* security model of attacker, where PPNs and Consumers follow the protocol but store all their inputs and actively try to deduce further information from the data.

Requirement 1. Obtaining the Producer's identities requires either: (1) finding the private key, (2) inverting the one-way function $E_{k_e}(y)$, or (3) a forward search by encrypting a set of predictable inputs. Regarding (1) and (2), we assume that the underlying Public Key Encryption (PKE) algorithm makes those attacks infeasible with the chosen key sizes. In order to prevent (3), the underlying PKE algorithm must be semantically secure. In Section VI we discuss a PKE algorithm with these characteristics. Moreover, thanks to properties 3 and 4 of Cryptosystem E_{k_e}, refreshing the random number w_p every α rounds ensures the pseudonym expiration and refreshment and limits its temporal validity.

Requirement 2. The identification number c of the monitoring Consumer is given as the input to the cryptosystem E_{k_e}. Therefore property 3 of the cryptosystem ensures that the pseudonyms are different even if the Producer's identity ID_p, the validity time window number $\lceil \frac{i}{\alpha} \rceil \alpha$ and the random number w_p are the same. Therefore, during the same pseudonym validity period, the same Producer is associated to a different pseudonym for every Consumer.

Requirement 3. Including the Producer's identity ID_p in the input of the pseudonym computation guarantees that two messages generated by distinct Producers are not mapped to the same output, even in case the monitoring Consumer c and the random number w_p are the same.

Requirement 4. Message 2 (SPECIFYMONITORINGSET) ensures that the Producer is aware about the set of Consumers which are monitoring his/her data.

Requirement 5. Messages 6 (RECOVERYREQUEST) and 7 (SENDIDENTITY) allow the Configurator to retrieve the Producer's identifier based on its pseudonym.

Note that, since the secret splitting scheme requires all the t shares to recover the secret, a collusion of less than t PPNs cannot recover Producers' measurements and associate them to the Producers' identities and/or pseudonyms. Therefore, even if the PPNs know the association between the Producers' identities and pseudonyms, they can relate them only to

the shares $s(x_i^p)$, but not to the measurement x_i^p itself. An additional attack scenario is given by a passive intruder which tries to collect multiple shares from a given Producer to recover the individual measurements. The assumption of a computationally secure confidential and authenticated channel between the nodes prevents this kind of attack.

Requirement 6. As the secret splitting scheme used to mask the metering measurements relies on the usage of multiple random numbers $q_1, q_2,...q_{t-1}$, the same measurement can be masked in different ways, according to the values of the random numbers. Conversely, two different measurements might originate shares having the same value, for some particular choices of the random numbers.

VI. Implementation of the Proposed Protocol

In this Section we propose an implementation of the ciphering scheme E_{k_e} and discuss possible attacks it may be subject to.

A. Pseudonyms Generation and ID Recovery

Our proposed implementation of E_{k_e} relies on RSA with the Optimal Asymmetric Encryption Padding (OAEP) cryptosystem [12, Cryptosystem 5.4]. Let $k_e = (b, e)$ and $k_d = (b, d)$ be the RSA keypair of the Configurator with modulus b, which is l bits long, and encryption and decryption exponents, respectively, e and d. The deterministic one-way functions

$$H_1 \colon \{0,1\}^{l-m-1} \to \{0,1\}^m$$

and

$$H_2 \colon \{0,1\}^{m} \to \{0,1\}^{l-m-1}$$

are systemwide masking generation functions (MGF), which can be implemented using the construction in PKCS#1 [13, Appendix B2]. The number m is a positive integer with $m < l < 2m$.

Let $E_{k_e} \colon \{0,1\}^m \times \{0,1\}^{l-m-1} \to \{0,1\}^l$ be defined as follows:

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

where

$$x_1 = x \oplus H_1(r)$$

which is m bits long, and

$$x_2 = r \oplus H_2(x_1)$$

which is $l - m$ bits long.

Recalling the definition from Section IV, the pseudonym of Producer p destined to Consumer c at round i is calculated as:

$$PD_c^p = E_{k_e}(ID_p \| c \| \lceil i/\alpha \rceil \alpha, w_p)$$

To obtain the Producer's identity, the Configurator computes the RSA decryption function

$$D_{k_d}(PD_c^p) = (PD_c^p)^d \bmod b = x_1 \| x_2$$

where $x_1 \in \{0,1\}^m$ and $x_2 \in \{0,1\}^{l-m-1}$. Then, it computes

$$ID_p \| c \| \lceil i/\alpha \rceil \alpha = H_1(x_2 \oplus H_2(x_1)) \oplus x_1$$

and eliminates $c \| \lceil i/\alpha \rceil \alpha$, which have a known length, to obtain ID_p.

The implementation described above has the properties described in Section IV, if RSA is implemented correctly and with an adequate key size. However, since the IDs and the round numbers are predictable the adversary can perform a forward search attack by simply encrypting the set of possible plaintext messages until the pseudonym PD_c^p is obtained [14]. Our protocol defies this attack by employing the RSA-OAEP construction [11], which adds randomness and structural constraints on the plaintext. The main difference to RSA-OAEP is that the same security nonce is re-used to calculate a Producer's pseudonym for different Consumers. Therefore an attacker could try to derive the pseudonym $PD_{c'}^p$, for a Consumer $c' \neq c$ given PD_c^p, exploiting the fact that several bits in the plaintext do not change. Consider, however, that RSA is a one way trapdoor permutation that is not XOR-malleable [15], meaning that knowing the XOR of the plaintexts gives information on the XOR between $PD_{c'}^p$ and PD_c^p with negligible probability. Therefore the probability of success of this attack is likewise negligible.

Finally, it is useful to discuss a suitable choice for the system parameters. Assuming assuming 128-bit long identifiers for Producers and Consumers, 64-bit long round numbers and 128-bit long nonces a suitable choice is $m = 512$ and $l = 1024$, which results in 1024-bit pseudonyms. These pseudonyms can be compressed to a shorter size with a deterministic one-way function if the application can tolerate that two IDs can map to the same pseudonym with negligible probability.

VII. Performance Assessment

In this section we evaluate the computational costs of the protocol presented in section IV. We define the number of operations required by each step as a function of the system parameters, i.e. number of Producers, PPNs and Consumers. We also consider the case of a user, i.e. a Producer or a Consumer, joining or leaving the system.

Every time a user joins or leaves the system, the *Setup* phase is re-executed and Π_c and Ψ_p are updated. In particular, if the new users are Consumers, they specify their Π_c to the Configurator, which checks the conformance of each request with cost $O(|C|)$. Then the Configurator computes Ψ_p with cost $O(|P|)$ and communicates it to the Producers. In case of new Producers joining or leaving the system, the same operations are performed with the same costs. During the following *Key Refresh* phase, the Configurator chooses his public key k_e and computes the private key k_d, with complexity $O(l^4)$.

The *Data Collection* phase is performed at every round i. For the computation of t shares $s(x_i^p, n)(n = 1, 2, .., t)$, the Producer p computes the measurements x_i^p, generates $t - 1$ integer random numbers and then calculates the t shares by simple subtractions. This operation has asymptotic complexity $O(|P| \cdot |N|)$. The largest number of operations are performed by the PPNs, which have to compute the pseudonyms PD_c^p

using cryptographically secure hash functions and RSA encryptions. The complexity is dominated by the RSA encryptions, which have complexity $O(l^2)$. The Consumer receives all the shares associated to different pseudonyms and, for each pseudonym, recovers the corresponding secret by simple summations. The complexity of this operation can be approximated by $O(|P| \cdot |N|)$.

The *Identity Recovery* phase, performed only in case of alarms/faults, involves a Consumer and the Configurator. The latter deciphers the pseudonym with his private key, exploiting the S&M algorithm, which has complexity $O(l^3)$. Then the Configurator obtains ID_p by removing $c \| \lceil i/\alpha \rceil \alpha$ and sends it to the Consumer. Table I summarizes the computational costs described above.

TABLE I
COMPUTATIONAL COSTS

Configurator	
Setup	$O(\|C\|) + O(\|P\|)$
KeyRefresh	$O(l^4)$
IDRecovery	$O(l^3)$
Producer	
DataCollection	$O(\|P\| \cdot \|N\|)$
PPN	
DataCollection	$O(\|P\| \cdot \|N\| \cdot \|C\| \cdot l^2)$
Consumer	
DataCollection	$O(\|P\| \cdot \|N\|)$

VIII. CONCLUSIONS

This paper proposes a pseudonymization protocol for smart metering measurements: data gathered by the smart meters can be collected by multiple utilities and third parties without revealing the association between users' identities and pseudonyms by means of a pseudonymization procedure performed at intermediate nodes called Privacy Preserving Nodes. The measurements generated by the meters are masked by means of a secret splitting scheme, so that the PPNs cannot have access to the metering data and relate them to the users' identities and/or pseudonyms. We also describe a possible implementation of the proposed algorithm and evaluate the security guarantees and performance the algorithm achieves.

REFERENCES

[1] National Institute of Standards and Technology (NIST), "Guidelines for smart grid cyber security," NIST Interagency Report 7628, Aug. 2010. [Online]. Available: http://www.nist.gov

[2] A. Cavoukian, J. Polonetsky, and C. Wolf, "Smartprivacy for the smart grid: embedding privacy into the design of electricity conservation," in *Identity in the Information Society*. Springer Netherlands, Apr. 2010.

[3] C. Rottondi, G. Verticale, and A. Capone, "A security framework for smart metering with multiple data consumers," in *1st IEEE INFOCOM CCSES Workshop on Green Networking and Smart Grids*, mar. 2012.

[4] C. Efthymiou and G. Kalogridis, "Smart grid privacy via anonymization of smart metering data," in *Smart Grid Communications (SmartGridComm), 2010 First IEEE International Conference on*, oct. 2010, pp. 238 –243.

[5] M. Jawurek, M. Johns, and K. Rieck, "Smart metering depseudonymization," in *Proceedings of the 27th Annual Computer Security Applications Conference*, ser. ACSAC '11. New York, NY, USA: ACM, 2011, pp. 227–236. [Online]. Available: http://doi.acm.org/10.1145/2076732.2076764

[6] A. Rial and G. Danezis, "Privacy-preserving smart metering," in *Proceedings of the 10th annual ACM workshop on Privacy in the electronic society*, ser. WPES '11. New York, NY, USA: ACM, 2011, pp. 49–60. [Online]. Available: http://doi.acm.org/10.1145/2046556.2046564

[7] Y. Zhang, W. Liu, and W. Lou, "Anonymous communications in mobile ad hoc networks," in *INFOCOM 2005. 24th Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings IEEE*, vol. 3, march 2005, pp. 1940 – 1951 vol. 3.

[8] H. D., "Pseudonym-based cryptography for anonymous communications in mobile ad hoc networks," in *Int. J. Security and Networks*, vol. Vol.2, Nos. 3/4. Arizona, AZ, USA: Interscience Enterprises Ltd, 2007, pp. 272–283.

[9] D. Henrici, J. Gotze, and P. Muller, "A hash-based pseudonymization infrastructure for rfid systems," in *Security, Privacy and Trust in Pervasive and Ubiquitous Computing, 2006. SecPerU 2006. Second International Workshop on*, june 2006, pp. 6 pp. –27.

[10] B. Riedl, T. Neubauer, G. Goluch, O. Boehm, G. Reinauer, and A. Krumboeck, "A secure architecture for the pseudonymization of medical data," in *Availability, Reliability and Security, 2007. ARES 2007. The Second International Conference on*, april 2007, pp. 318 –324.

[11] M. Bellare and P. Rogaway, "Optimal asymmetric encryption," in *Advances in Cryptology EUROCRYPT'94*, ser. Lecture Notes in Computer Science, A. De Santis, Ed. Springer Berlin / Heidelberg, 1995, vol. 950, pp. 92–111.

[12] D. Stinson, *Cryptography Theory and Practice, Second Edition*. CRC Press, 2005.

[13] K. B. Jonsson J., "Public-key cryptography standards (PKCS) #1: Rsa cryptography, specifications version 2.1," 2003.

[14] S. A. V. Alfred J. Menezes, Paul C. van Oorschot, *Handbook of applied cryptography*. CRC Press, 1996.

[15] V. Shoup, "Oaep reconsidered," Cryptology ePrint Archive, Report 2000/060, 2000, http://eprint.iacr.org/.