MUSP: A Multi-service, User Self-controllable and Privacy-preserving System for Smart Metering

DOI:
10.1109/ICC.2015.7248418

Document Version
Accepted author manuscript

Citation for published version (APA):
Mustafa, M. A., Zhang, N., Kalogridis, G., & Fan, Z. (2015). MUSP: A Multi-service, User Self-controllable and Privacy-preserving System for Smart Metering. In *2015 IEEE International Conference on Communications (ICC)* IEEE. https://doi.org/10.1109/ICC.2015.7248418

Published in:
2015 IEEE International Conference on Communications (ICC)

MUSP: Multi-service, User Self-controllable and Privacy-preserving System for Smart Metering

Mustafa A. Mustafa and Ning Zhang
School of Computer Science
The University of Manchester
Oxford Road, Manchester, M13 9PL, UK
Email: {mustafm, nzhang}@cs.man.ac.uk

Georgios Kalogridis and Zhong Fan
Toshiba Research Europe Limited
Telecommunications Research Laboratory
32 Queen Square, Bristol, BS1 4ND, UK
Email: {george, zhong.fan}@toshiba-trel.com

Abstract—**This paper proposes a Multi-service, User Self-controllable and Privacy-preserving (MUSP) system for secure smart metering. This system has a number of novel properties. Firstly, it can report users' fine-grained consumption data to grid operators and suppliers securely and with user privacy preservation capability. These are achieved by using a homomorphic encryption technique in conjunction with selective data aggregation and distribution methods, so only the aggregated data are delivered to the authorised data recipients only on a need-to-know basis. Secondly, it allows suppliers to access their customers' attributable meter readings regularly. To protect users' privacy, suppliers, by default, can access new data only at a low frequency (e.g. once a month). However, MUSP allows users (1) to adjust (control) this frequency and (2) to release new data by demand (e.g. when change of tariff occurs), thus putting users' privacy preservation in their own hands. Thirdly, it is equipped with an easy and user friendly supplier switching facility to allow users to switch providers easily and conveniently. Security analysis and performance evaluation demonstrate that the MUSP system can protect users' privacy while providing these services in an efficient and scalable manner.**

I. Introduction

Advanced Metering Infrastructure (AMI) is part of the emerging Smart Grid (SG) that supports metering data communications [1], [2]. An important component of AMI is Smart Meters (SMs). SMs are devices capable of measuring, recording and communicating users' detailed electricity consumption data to different SG entities such as grid operators and suppliers. By making use of such data, these entities would be able to operate the SG more efficiently and reliably.

However, along with the benefits, the SG also introduces new security and privacy issues [2]–[8]. For example, SG entities having access to detailed consumption data may use non-intrusive load monitoring techniques to build individual users' electricity consumption patterns, thus breaching users' privacy [9]. To prevent such breaches, none of the SG entities should be allowed to access individual users' metering data at a high frequency. However, to allow account management, suppliers must access their customers' (individual users') attributable metering data[1]. Also, some users may not have stringent privacy requirements; they may not mind their suppliers to access their attributable metering data more frequently in return for incentives. So, a metering system should be flexible (able to support diverse requirements). One way to facilitate this flexibility is to allow users to control the release frequency of their attributable data. As electricity markets are liberalized (users are free to choose their suppliers), the system should also allow users to switch among different suppliers easily and securely. Thus, a smart metering system should support the following services in a secure, privacy-preserving and efficient manner:

(S1) Report aggregated consumption data to operators/suppliers at a high frequency (for efficient grid management);
(S2) Report meter readings to its user at a high frequency (for making the user aware of his/her electricity usage);
(S3) Allow suppliers to access attributable meter readings without users' intervention (for account management);
(S4) Allow users to adjust the release frequency of their attributable meter readings (for privacy level control);
(S5) Allow users to release their attributable meter readings on demand (for accommodating cases like tariff change);
(S6) Allow users to easily switch among different suppliers.

This paper proposes such a system, a novel Multi-service, User Self-controllable and Privacy-preserving (MUSP) smart metering system. MUSP utilizes a homomorphic encryption technique to aggregate users' detailed consumption data and only distribute the aggregated data to the respective grid operators and suppliers on a need-to-know basis. The latter property requires that users' data are grouped depending on their intended recipients and that the homomorphic encryption is applied accordingly. MUSP also, with the help of a trusted party, allows users (1) to adjust (control) the frequency with which their suppliers can access their new attributable meter readings and (2) to securely switch suppliers. To the best of our knowledge, this is the first work on designing a smart metering system that supports all of the above mentioned services.

The rest of the paper is organized as follows: §II reviews related work. §III gives the design preliminaries. §IV describes our novel MUSP system, §V its security analysis and §VI its performance evaluation. §VII concludes the paper.

II. Related Work

Efthymiou et al. [10] proposed a method for anonymizing users' metering data sent at high frequency, so data recipients cannot link data to their owners. Lin et al. [11] proposed a system to allow users' data to be accessed at multiple time

[1] Attributable metering data: data that are tied to an individual SM/user.

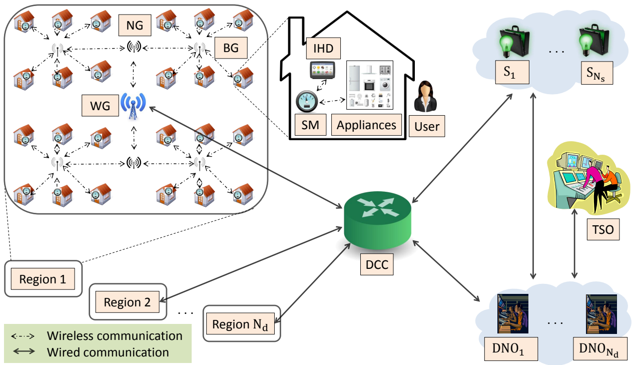

Fig. 1. The conceptual architecture of AMI in smart grid.

granularities. Mármol et al. [12] proposed a system whereby a supplier aggregates the encrypted data sent by users and uses an aggregated key sent by a key aggregator to recover the aggregated data. Although these solutions preserve users' privacy, they impose high levels of communication overheads.

To reduce the communication overheads, individual users' data should be aggregated and only the aggregated data be delivered to the final recipients. To preserve users' privacy during a data aggregation process, homomorphic encryption based aggregation schemes have been proposed [13]–[18]. However, these schemes use a single-recipient system model, thus they are not efficient when applied to liberalized electricity markets where multiple recipients are expected to access the aggregated data of different sets of users.

To address this limitation, Rottondi et al. [19] proposed an architecture with additional functional entities, Privacy Preserving Nodes (PPNs), that collect users' data encrypted by a Secret Sharing Scheme (SSS) and perform aggregation of different sets of the data for their intended recipients governed by access rights. This solution has two drawbacks: (1) increased SG complexity due to the use of PPNs, and (2) increased communication overheads due to the use of SSS. The first drawback was overcome by allocating the tasks of PPNs to existing SG nodes (gateways) [20], and the second one by replacing SSS with homomorphic encryption [21]. However, the schemes do not (1) support the collection of users' attributable metering data, (2) allow users to control their privacy preservation levels and (3) support supplier switch. This paper extends the work in [21] to address these limitations by designing MUSP that supports the services listed in Section I.

III. Design Preliminaries

A. System Model

The system model, shown in Fig. 1, consists of:

- Grid operators: one Transmission System Operator (TSO) responsible for balancing the entire grid and N_d Distribution Network Operators (DNOs) each responsible for maintaining distribution networks in a particular region.
- Suppliers (Ses): N_s suppliers each responsible for supplying electricity to a subset of users (i.e. its customers).
- Smart Meter (SM): an advanced metering device that measures its user's electricity usage on per timeslot basis.

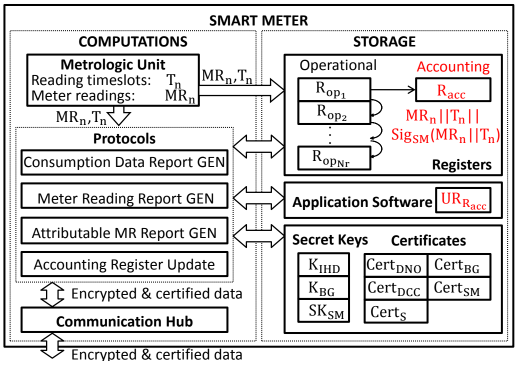

Fig. 2. The proposed SM architecture.

- In-Home Display (IHD): a device linked to an SM and used by the user to access, update or share metering data.
- Data Communication Company (DCC): it collects, (aggregates) and communicates metering data to SG entities.
- Networking facility: a hierarchical network facility comprised of Building area network Gateways (BGs), Neighbourhood area network Gateways (NGs) and Wide area network Gateways (WGs). It connects SMs to the DCC.

B. Assumptions

- Metering data collected for operational purposes do not have to be attributable to users. Aggregated data is sufficient if it is authentic and tied to a region and supplier.
- Aggregated data are collected at a high frequency, e.g. every 30 minutes, to enable efficient grid management.
- Metering data collected for account management purposes must be attributable to a particular SM/user.
- Attributable metering data are collected at a low frequency (e.g. monthly) and on demand (e.g. tariff change).

C. Threat Model

- SMs are tamper-proof. It is hard for entities (including their users) to tamper with them successfully.
- DCC, DNOs, TSO and suppliers are honest-but-curious. They follow protocol specifications but may try to find out confidential data of users or (other) operators/suppliers.
- External entities are not trustworthy. They may intercept data in transit trying to access confidential (private) data.

D. Smart Meter Architecture

MUSP introduces a single functional component, an accounting register R_{acc}, into the existing SM architecture (Fig. 2). Different from operational registers, $R_{op_1}, \ldots, R_{op_{Nr}}$, which are used for storing raw metering data, R_{acc} stores user's attributable metering data. Each operational register stores Meter Reading (MR) and timeslot, e.g. $\{MR_n \parallel T_n\}$, but R_{acc} also stores the SM's digital signature on the data, e.g. $\{MR_n \parallel T_n \parallel Sig_{SM}(MR_n \parallel T_n)\}$. $R_{op_1}, \ldots, R_{op_{Nr}}$ are updated per timeslot when the content of R_{op_i} is shifted to $R_{op_{i+1}}$ and the freshest $\{MR_n \parallel T_n\}$ is stored in R_{op_1}, thus R_{op_1} ($R_{op_{Nr}}$) stores the most (least) recent data available on the SM. R_{acc} is regularly updated with data from R_{op_1} and the update rate ($UR_{R_{acc}}$) is used as an input value of the application software.

Each SM is installed with the following secret keys: a symmetric key shared between the SM and its IHD (K_{IHD}), a symmetric key shared between the SM and its local BG (K_{BG}) and its signature signing key (SK_{SM}). It is also loaded with the certificates of the regional DNO ($Cert_{DNO}$), the DCC ($Cert_{DCC}$), its user's supplier ($Cert_{S}$), the local BG ($Cert_{BG}$) and its own certificate ($Cert_{SM}$). Each certificate contains the owner's ID and public key. Each DNO's certificate also contains the DNO's homomorphic public key. In this work, we use the Paillier Cryptosystem [22] as it has an additive homomorphism property and is relatively efficient and semantically secure. Information about homomorphic encryption (including Paillier) schemes can be found in [23].

IV. The MUSP System

The MUSP system is operated by five protocols used for: (a) aggregated consumption data reporting to SG entities, (b) MR reporting to users, (c) attributable MR reporting to suppliers, (d) accounting register update and (e) supplier switching.

A. Aggregated Consumption Data Reporting to SG Entities

This protocol is designed to distribute users' aggregated consumption data to authorised SG entities on a need-to-know basis at a high frequency. It is invoked at the beginning of each timeslot, and each protocol message contains the IDs of the message originator and recipient, timestamps along with the originator's signature on the message. The message, upon reception, is verified by the recipient. For simplicity, the description below omits these details, but rather it focuses on the measures taken to ensure secure aggregation of users' consumption data and secure reporting the aggregated data to the need-to-know SG entities (grid operators and suppliers). The protocol involves six major steps described next.

1) Consumption data report GEN (by SMs): Each SM, say SM_i, generates a report containing its user's consumption data for the past timeslot and sends it to its local BG, i.e. SM_i

a) reads MR_n and $MR_{n\text{-}1}$ from registers R_{op_1} and R_{op_2}, respectively, and calculates the user's consumption data, $CD_n = MR_n - MR_{n\text{-}1}$, for the past timeslot;

b) generates a random number, r_n;

c) encrypts CD_n using its regional DNO's homomorphic public key and r_n (used as an input value to the encryption algorithm too), i.e. $C_{SM_i} = Enc(HPK_{DNO_j}, CD_n, r_n)$; this is to resist passive attacks by external entities;

d) appends the user's supplier's ID to C_{SM_i} and encrypts the result using K_{BG}, i.e. $c_{SM_i} = E(K_{BG}, \{ID_{S_u} \parallel C_{SM_i}\})$; this is to resist passive attacks by authorised entities;

e) appends the ID of the regional DNO to c_{SM_i} and sends the result, i.e. $M_{SM_i} = \{ID_{DNO_j} \parallel c_{SM_i}\}$, to the local BG.

2) Supplier-based data aggregation (by gateways): Each gateway, say BG_i, aggregates the respective sets of received ciphertexts based on the suppliers' IDs attached to them, i.e. upon the receipt of a message M_{SM_i}, BG_i

a) decrypts c_{SM_i}, i.e. $\{ID_{S_u} \parallel C_{SM_i}\} = D(K_{BG}, c_{SM_i})$;

b) groups the ciphertexts with the same supplier ID;

c) aggregates the ciphertexts in each group to form a single aggregated ciphertext for the group, i.e. $C_{BG_i,S_u} = \prod C_{SM}$, and appends the corresponding supplier's ID to C_{BG_i,S_u};

d) constructs a message containing regional DNO's ID and aggregated ciphertexts each appended with the corresponding supplier's ID, $M_{BG_i} = \{ID_{DNO_j} \parallel (ID_{S_1} \parallel C_{BG_i,S_1}), \ldots, (ID_{S_{N_s}} \parallel C_{BG_i,S_{N_s}})\}$, and sends M_{BG_i} to the local NG.

Each NG/WG does similar operations as those carried out by each BG except that it processes messages from BGs/NGs and skips the decryption step as gateways do not encrypt data.

3) Region-supplier-based data aggregation and distribution (by DCC): DCC, based on the DNOs' and suppliers' IDs contained in the received messages, aggregates the ciphertexts contained in the messages, and distributes the aggregated ciphertexts to the respective DNOs/suppliers. In detail, DCC

a) sorts the received messages containing the same DNO ID into one group, thus forming N_d different groups;

b) sorts the ciphertexts contained in the messages in each group into subgroups based on the supplier's ID attached to each, thus forms N_s different subgroups in each group;

c) aggregates the ciphertexts in each subgroup, e.g. $C_{DNO_j,S_u} = \prod C_{WG,S_u}$, thus forming ($N_d$ x N_s) different aggregated ciphertexts, where C_{DNO_j,S_u} is the aggregated ciphertext of all the users located in the region operated by DNO_j and supplied by S_u;

d) constructs a message for each DNO, that contains the supplier-based aggregated ciphertexts of the users living in the DNO's region, e.g. $M_{DCC,DNO_j} = \{(ID_{S_1} \parallel C_{DNO_j,S_1}), \ldots, (ID_{S_{N_s}} \parallel C_{DNO_j,S_{N_s}})\}$, and sends it to the DNO;

e) constructs a message for each supplier, which contains the region-based aggregated ciphertexts of all the supplier's customers, e.g. $M_{DCC,S_u} = \{(ID_{DNO_1} \parallel C_{DNO_1,S_u}), \ldots, (ID_{DNO_{N_d}} \parallel C_{DNO_{N_d},S_u})\}$, and sends it to the supplier. Note that suppliers cannot decrypt these ciphertexts as they do not know the corresponding homomorphic private keys. These ciphertexts are used by the suppliers for data verifications (more details in Section IV-A5).

4) Region-supplier-based data recovery (by DNOs): DNOs recover aggregated consumption data (ACD) and send sets of data to the corresponding suppliers and TSO through secure channels. For this, each DNO, say DNO_j, does the following:

a) for each supplier, e.g. S_u, it

   i. recovers the ACD by decrypting the respective aggregated ciphertext using its homomorphic private key, i.e. $ACD_{DNO_j,S_u} = Dec(HSK_{DNO_j}, C_{DNO_j,S_u})$,

   ii. recovers the random number embedded in C_{DNO_j,S_u}, i.e. $r_{DNO_j,S_u} = Rnr(HSK_{DNO_j}, C_{DNO_j,S_u}, ACD_{DNO_j,S_u})$,

   iii. constructs a message $M_{DNO_j,S_u} = \{r_{DNO_j,S_u} \parallel ACD_{DNO_j,S_u}\}$ and sends it to the supplier;

b) sums all the ACD recovered, i.e. $M_{DNO_j,TSO} = \sum_{u=1}^{N_s} ACD_{DNO_j,S_u}$, and sends the result to the TSO.

5) Supplier-region-based data access and ciphertext-based data verification (by suppliers): Suppliers access and verify the ACD of their customers. Each supplier, e.g. S_u,

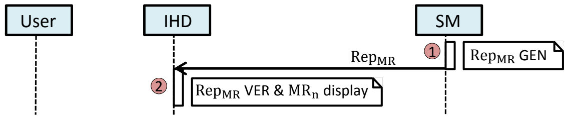
User
IHD
SM
Rep_{MR}
1
Rep_{MR} GEN
2
Rep_{MR} VER & MR_n display

Fig. 3. An outline of the meter reading reporting protocol.

a) obtains the message, $\text{M}_{\text{DCC},\text{S}_\text{u}} = \{(\text{ID}_{\text{DNO}_1} \parallel \text{C}_{\text{DNO}_1,\text{S}_\text{u}}), \ldots, (\text{ID}_{\text{DNO}_{\text{N}_\text{d}}} \parallel \text{C}_{\text{DNO}_{\text{N}_\text{d}},\text{S}_\text{u}})\}$, sent from the DCC;

b) for each DNO, e.g. DNO_j, it

   i. obtains $\text{M}_{\text{DNO}_\text{j},\text{S}_\text{u}} = \{\text{r}_{\text{DNO}_\text{j},\text{S}_\text{u}} \parallel \text{ACD}_{\text{DNO}_\text{j},\text{S}_\text{u}}\}$,

   ii. encrypts $\text{ACD}_{\text{DNO}_\text{j},\text{S}_\text{u}}$ using the homomorphic public key of DNO_j and $\text{r}_{\text{DNO}_\text{j},\text{S}_\text{u}}$, i.e. $\text{C'}_{\text{DNO}_\text{j},\text{S}_\text{u}} = \text{Enc}(\text{HPK}_{\text{DNO}_\text{j}}, \text{ACD}_{\text{DNO}_\text{j},\text{S}_\text{u}}, \text{r}_{\text{DNO}_\text{j},\text{S}_\text{u}})$,

   iii. verifies and accepts $\text{ACD}_{\text{DNO}_\text{j},\text{S}_\text{u}}$ if $\text{C}_{\text{DNO}_\text{j},\text{S}_\text{u}}$ received in step a) is identical to $\text{C'}_{\text{DNO}_\text{j},\text{S}_\text{u}}$ computed above;

c) sums the verified ACD, $\text{ACD}_{\text{S}_\text{u}} = \sum_{\text{j}=1}^{\text{N}_\text{d}} \text{ACD}_{\text{DNO}_\text{j},\text{S}_\text{u}}$.

6) Region-based data access (by TSO): Upon receiving the ACDs from all the DNOs, TSO sums the ACDs, $\text{ACD}_{\text{TSO}} = \sum_{\text{j}=1}^{\text{N}_\text{d}} \text{ACD}_{\text{DNO}_\text{j}}$, to get the ACD of all the users in the grid.

B. Meter Reading Reporting to Users

MRs can be used for improving electricity usage awareness too by making them available to their respective users at a very high frequency, e.g. once every 30 sec. As shown in Fig. 3, this protocol has one message: a MR Report (Rep_{MR}).

1) Rep_{MR} *GEN:* SM generates and sends Rep_{MR} to its IHD.

In detail, SM reads the current MR, MR_n, from its metrologic unit, encrypts it using the key shared with the IHD (K_{IHD}) to generate $\text{C}_{\text{IHD}} = \text{E}(\text{K}_{\text{IHD}}, \text{MR}_\text{n})$, and generates a keyed-hash message authentication code (HMAC) on C_{IHD} also using K_{IHD}, i.e. $\text{H}_{\text{K}_{\text{IHD}}}(\text{C}_{\text{IHD}})$. It then sends the report, $\text{Rep}_{\text{MR}} = \{\text{C}_{\text{IHD}} \parallel \text{H}_{\text{K}_{\text{IHD}}}(\text{C}_{\text{IHD}})\}$, to its IHD.

2) Rep_{MR} *VER &* MR_n *display:* The IHD verifies Rep_{MR} and recovers MR_n using K_{IHD}. It then displays (and stores) MR_n. The user can access MR_n via IHD which is assumed to be in a physically secure place (inside the user's house).

C. Attributable Meter Reading Reporting to Suppliers

Attributable MRs are used for accounting (billing) purposes and released upon suppliers' requests. As shown in Fig. 4, this protocol has two messages: an attributable MR Request ($\text{Req}_{\text{attr.MR}}$) and an attributable MR Report ($\text{Rep}_{\text{attr.MR}}$).

1) $\text{Req}_{\text{attr.MR}}$ *GEN:* A supplier, S, generates and dispatches $\text{Req}_{\text{attr.MR}}$ to an SM of its customer.

S generates randomly a session key (K), encrypts K using the SM's public key, $\text{C}_\text{S} = \text{Enc}(\text{PK}_{\text{SM}}, \text{K})$, and signs the result using its private key to generate a request, $\text{Req}_{\text{attr.MR}} = \{\text{C}_\text{S} \parallel \text{Sig}_\text{S}(\text{C}_\text{S})\}$. It then sends the request to the SM and starts a count-down timer that defines a timeslot in which S expects to receive a report from the SM. This measure is taken to resist denial-of-service attacks launched by malicious SMs.

2) $\text{Req}_{\text{attr.MR}}$ *VER &* $\text{Rep}_{\text{attr.MR}}$ *GEN:* SM verifies $\text{Req}_{\text{attr.MR}}$, generates $\text{Rep}_{\text{attr.MR}}$ and sends it to S.

Upon the receipt of $\text{Req}_{\text{attr.MR}}$, SM verifies its authenticity using the public key contained in the certificate of S, Cert_S, stored on SM, and recovers K using its private key, SK_{SM}. If the request is authentic, SM then reads the data, $\text{Data} = \{\text{MR}_\text{n} \parallel \text{T}_\text{n} \parallel \text{Sig}_{\text{SM}}(\text{MR}_\text{n} \parallel \text{T}_\text{n})\}$, from its R_{acc}, encrypts the data using K, i.e. $\text{C}_{\text{SM}} = \text{E}(\text{K}, \text{Data})$, and generates an HMAC on C_{SM} also using K, i.e. $\text{H}_\text{K}(\text{C}_{\text{SM}})$. It sends the report, $\text{Rep}_{\text{attr.MR}} = \{\text{C}_{\text{SM}} \parallel \text{H}_\text{K}(\text{C}_{\text{SM}})\}$, to S.

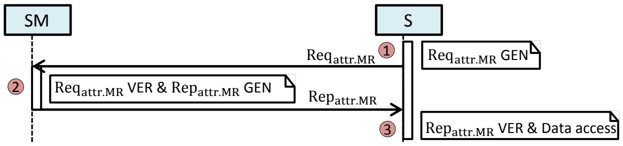
SM
S
$\text{Req}_{\text{attr.MR}}$
1
$\text{Req}_{\text{attr.MR}}$ GEN
2
$\text{Req}_{\text{attr.MR}}$ VER & $\text{Rep}_{\text{attr.MR}}$ GEN
$\text{Rep}_{\text{attr.MR}}$
3
$\text{Rep}_{\text{attr.MR}}$ VER & Data access

Fig. 4. An outline of the attributable meter reading reporting protocol.

3) $\text{Rep}_{\text{attr.MR}}$ *VER &* Data *access:* S verifies $\text{Rep}_{\text{attr.MR}}$ and recovers Data using K. Note that Data contains attributable MR, i.e. MR timestamped and signed by the user's SM.

It is worth noting that the approach used in this protocol *simplifies* data access management at SMs as an SM responds to every $\text{Req}_{\text{attr.MR}}$ sent by its user's supplier and the generated report, $\text{Rep}_{\text{attr.MR}}$, carries the same data until R_{acc} is updated regardless of the time and frequency of $\text{Req}_{\text{attr.MR}}$s' arrivals.

D. Accounting Register Update

To balance privacy and functionality, MUSP is equipped with three ways of controlling the release of new attributable MRs to a supplier (via updating R_{acc} in an SM): R_{acc} Update Rate ($\text{UR}_{\text{R}_{\text{acc}}}$) set by manufacturer, $\text{UR}_{\text{R}_{\text{acc}}}$ set by user and R_{acc} update on demand.

1) $\text{UR}_{\text{R}_{\text{acc}}}$ *set by manufacturer:* To ensure the basic functionality of the system with maximum protection of a user's privacy, we should have a minimum $\text{UR}_{\text{R}_{\text{acc}}}$ value set just sufficient for a supplier to accomplish its basic duty - generating accurate bills regularly. This value should be (i) agreed among regulators, suppliers and users, (ii) used as the default value and (iii) set by the SM manufacturer. For example, this value can be set to 'once per month', meaning that, by default, SM updates its R_{acc} only once per month, so the supplier can access the user's new attributable MRs once every month.

2) $\text{UR}_{\text{R}_{\text{acc}}}$ *set by user:* In addition to the minimum $\text{UR}_{\text{R}_{\text{acc}}}$ value, there should also be a facility to allow an SM user to set $\text{UR}_{\text{R}_{\text{acc}}}$ to a higher value, e.g. 'once a day'. Obviously, the higher the $\text{UR}_{\text{R}_{\text{acc}}}$ value, the more frequently the supplier can pull new attributable MRs from the SM, and the lower the privacy protection level. As depicted in Fig. 5, this facility is initiated by the user and notifies the supplier and DCC. Also, each message should be encrypted with the intended recipient's public key and signed with the sender's private key.

a) A user (User) and her supplier (S) establish a secure channel through which User requests a new $\text{UR}_{\text{R}_{\text{acc}}}$ value.

b) S sends a message to DCC, and the message contains the user's SM ID (ID_{SM}) and the requested $\text{UR}_{\text{R}_{\text{acc}}}$ value.

c) DCC receives the message, generates a unique user code (U_{code}) and sends a message containing U_{code} and the new $\text{UR}_{\text{R}_{\text{acc}}}$ value to the user's IHD (via the user's SM).

d) User's IHD receives the message sent by DCC and displays U_{code} and the new $\text{UR}_{\text{R}_{\text{acc}}}$ value.

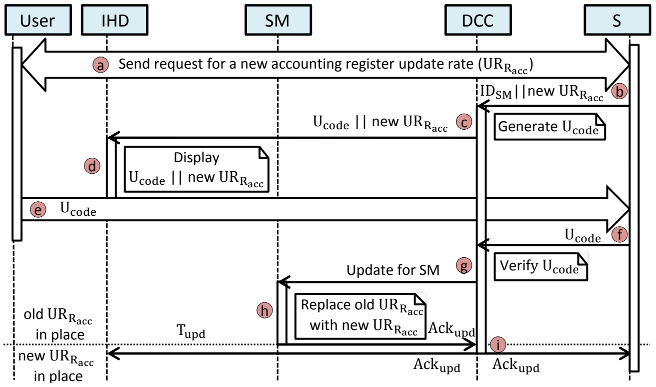

Fig. 5. An outline of the accounting register UR-change protocol.

e) User verifies that the $UR_{R_{acc}}$ value displayed is the same as the value initially requested, and sends U_{code} to S through a new or already established secure channel.
f) S receives U_{code} and forwards it to DCC.
g) DCC receives U_{code} sent by S, verifies that U_{code} received is the same as U_{code} sent to User, and sends an update to SM containing the new $UR_{R_{acc}}$ value.
h) SM replaces the existing $UR_{R_{acc}}$ value used as input into its application software with the new $UR_{R_{acc}}$ value and sends an acknowledgement (Ack_{upd}) to DCC.
i) DCC forwards Ack_{upd} to IHD and S.

Note that as the supplier is aware of the $UR_{R_{acc}}$ value, it does not need to be notified when a scheduled R_{acc} update occurs, so MUSP introduces a low level of communication overheads. It also supports *smart report timing* as the supplier can schedule the times it sends requests to its customers' SMs according to their R_{acc} update rates, and still obtain the attributable MRs measured at the same time from all its customers.

3) R_{acc} *update on demand:* There are times when a supplier may need to access new attributable MRs on demand (outside the schedule times), e.g. when there is a tariff/account holder change. In such cases, a new attributable MR release should be initiated by the user (as this should be approved by the user on per release basis) and the supplier should be notified of this release. Note that such updates will not affect the regular R_{acc} update times set beforehand. The protocol is shown in Fig. 6.

a) User initiates the update on demand protocol by choosing the corresponding option from the menu of her IHD. This option could also be username/password protected.
b) IHD generates, encrypts and integrity protects (using K_{IHD}) an update command (Cmd_{upd}) and sends it to SM.
c) SM verifies Cmd_{upd}, updates its R_{acc} and sends an encrypted and signed notification ($Notif_{upd}$) to S.

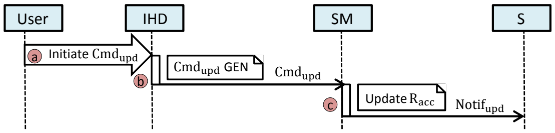

Fig. 6. An outline of the update-on-demand protocol.

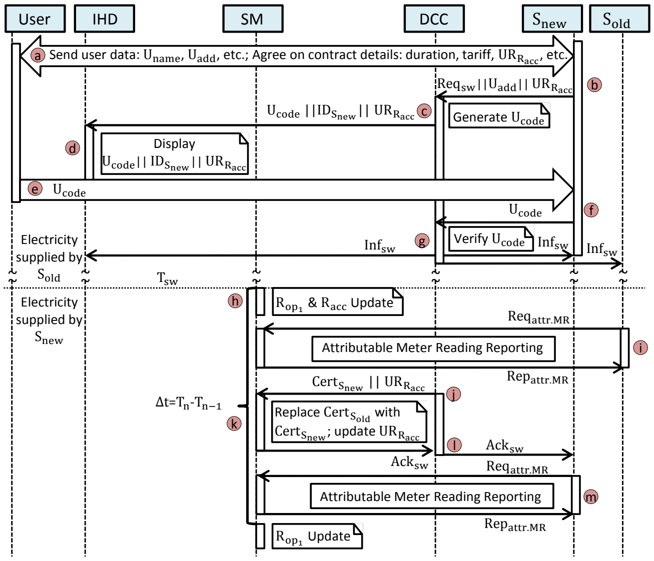
User
IHD
SM
DCC
S_{new}
S_{old}
a Send user data: U_{name}, U_{add}, etc.; Agree on contract details: duration, tariff, $UR_{R_{acc}}$, etc.
Req_{sw}||U_{add}|| $UR_{R_{acc}}$ b
U_{code} ||$ID_{S_{new}}$|| $UR_{R_{acc}}$ c
Generate U_{code}
d
Display
U_{code}|| $ID_{S_{new}}$|| $UR_{R_{acc}}$
e U_{code}
U_{code} f
Electricity
supplied by
S_{old}
Inf_{sw} g
Verify U_{code}
Inf_{sw}
Inf_{sw}
T_{sw}
Electricity
supplied by
S_{new}
h
R_{op_1} & R_{acc} Update
$Req_{attr.MR}$
Attributable Meter Reading Reporting
i
$Rep_{attr.MR}$
$Cert_{S_{new}}$ || $UR_{R_{acc}}$ j
$\Delta t=T_n-T_{n-1}$
k
Replace $Cert_{S_{old}}$ with
$Cert_{S_{new}}$; update $UR_{R_{acc}}$
l
Ack_{sw}
Ack_{sw}
$Req_{attr.MR}$
Attributable Meter Reading Reporting
m
$Rep_{attr.MR}$
R_{op_1} Update

Fig. 7. An outline of the supplier-switching protocol.

E. Supplier Switch

MUSP also allows users to switch suppliers easily and securely. The protocol supporting this is outlined in Fig. 7.

a) A user (User) and a new supplier (S_{new}) establish a secure channel through which User sends S_{new} her data, including name (U_{name}), address (U_{add}), any contract details, e.g. binding duration, tariff, $UR_{R_{acc}}$ value.
b) S_{new} sends a message to DCC containing the request to switch supplier (Req_{sw}), U_{add} and the agreed $UR_{R_{acc}}$ value.
c) DCC receives the message, identifies the user's SM based on U_{add}, generates a unique user code (U_{code}) and sends a response to the user's IHD containing U_{code}, S_{new}'s ID ($ID_{S_{new}}$) and $UR_{R_{acc}}$ value.
d) IHD displays the response: $U_{code} \parallel ID_{S_{new}} \parallel UR_{R_{acc}}$.
e) User verifies (1) $ID_{S_{new}}$ displayed on IHD is the ID of S_{new} and (2) $UR_{R_{acc}}$ displayed on IHD is the same as $UR_{R_{acc}}$ initially agreed with S_{new}, and sends U_{code} to S_{new} through a new or already established secure channel.
f) S_{new} receives U_{code} and forwards it to DCC.
g) DCC receives U_{code}, verifies that U_{code} is the same as U_{code} sent to User, and sends the user's IHD (via SM), current supplier (S_{old}) and S_{new} information about the switch (Inf_{sw}) including the date/time of the switch (T_{sw}).
h) SM updates its registers (incl. R_{acc}) at T_{sw} as scheduled.
i) S_{old} pulls the (final) MR from SM using the attributable MR reporting as in Section IV-C. This is successful as SM still stores its user's current supplier's certificate ($Cert_{S_{old}}$).
j) DCC sends an update to SM containing the certificate of S_{new} ($Cert_{S_{new}}$) and the (new) $UR_{R_{acc}}$ value.
k) SM replaces (1) $Cert_{S_{old}}$ with $Cert_{S_{new}}$ on its storage and (2) the existing $UR_{R_{acc}}$ value with the new $UR_{R_{acc}}$ value, and sends an acknowledgement (Ack_{sw}) to DCC.
l) DCC receives and forwards Ack_{sw} to S_{new}.
m) S_{new} pulls the (initial) MR from SM successfully using the attributable MR reporting as SM now stores $Cert_{S_{new}}$.

Note that the supplier switching process is easy and simplified as only the certificate of the current supplier stored on an SM needs to be replaced with the certificate of the new supplier. No change of cryptographic secret keys is required.

V. Security Analysis

A. Aggregated Consumption Data Reporting

Each individual user's consumption data is double encrypted (first with the regional DNO's homomorphic public key, then with the key shared between the user's SM and its local BG, i.e. K_{BG}) while in transit between the SM and BG, thus (unlike other aggregating schemes) making our protocol resistant to passive eavesdropping attacks by both outsiders and authorised insiders. Each authorised insider (i.e. DNOs, suppliers, the TSO) can only access the aggregated data of their respective users. Assuming that the received aggregated data contains the individual data of sufficient number of users, such that it is difficult for the data recipients to decompose the aggregated data into individual users' data, users' privacy is protected.

B. Meter Reading Reporting

MRs from an SM are encrypted and integrity protected using a key, $\mathrm{K}_{\mathrm{IHD}}$, shared only between the SM and its IHD. So, only the users who have access to the IHD could access the MRs. Assuming that the IHD is located in a physically secure location (inside the user's house), only authorised users can access MRs at a high frequency, thus preserving users' privacy. Obviously, accessing IHD can also be username/password controlled, ensuring that no user, rather than the one knowing the password (i.e. account holder), can gain access to the MRs.

C. Attributable Meter Reading Reporting

Attributable MRs from $\mathrm{R}_{\mathrm{acc}}$ of an SM are encrypted and integrity protected using a session key, K, shared only between the SM and its user's supplier, so only the user's supplier can access the user's attributable MRs. Moreover, both the SM and the supplier are resistant to denial-of-service attacks as the SM only accepts requests sent by its user's supplier and the supplier only accepts reports sent by the SM to which requests have been sent. However, as the supplier is allowed to obtain data from $\mathrm{R}_{\mathrm{acc}}$ as many times as it wants, to protect the user's privacy, the content of $\mathrm{R}_{\mathrm{acc}}$ should be updated at a low frequency. This can be achieved by setting $\mathrm{UR}_{\mathrm{R}_{\mathrm{acc}}}$ to a low value, e.g. one update per month, by default. This setting will assure the supplier that it can access its user's new attributable MRs, at least once a month, without the user's involvement. Also, having $\mathrm{R}_{\mathrm{acc}}$ allows users to adjust their own $\mathrm{UR}_{\mathrm{R}_{\mathrm{acc}}}$ values, thus managing their own privacy protection levels.

D. Updates to SMs

Any change made to $\mathrm{UR}_{\mathrm{R}_{\mathrm{acc}}}$ (or a supplier switch) is verified by the user with the help of a code generated by a trusted third party, i.e. DCC. As long as DCC generates an unique user code every time it receives a request made by a supplier, DCC can be assured that the request is indeed from a legitimate user.

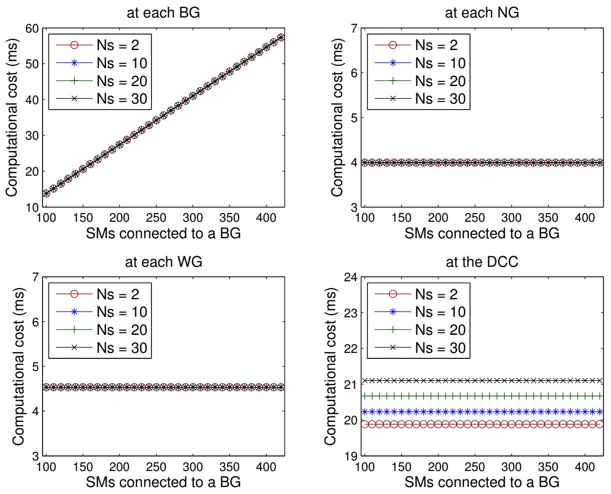

Fig. 8. Computational cost of the aggregated data reporting protocol.

VI. Performance Evaluation

This section evaluates the performance of the MUSP system. The computationally expensive operations used in the MUSP protocols are asymmetric encryption/decryption and digital signature generation/verification.

A. Aggregated Consumption Data Reporting

During each timeslot, an SM performs one asymmetric encryption and sends one message (thus it generates one digital signature) to its local BG, a gateway verifies one message per child entity and sends one message (thus it generates one digital signature) to its parent entity, and DCC verifies one message per child WG and sends one message to each DNO and each supplier (thus it generates $\mathrm{N}_{\mathrm{d}}+\mathrm{N}_{\mathrm{s}}$ digital signatures).

We ran experiments with pbc [24] and miracl [25] libraries on 3.0 GHz-processor and 4 GB-memory machine to study the operational costs. Our results indicate that asymmetric (homomorphic) encryption (i.e. Paillier encyption with $|n^2| = 2,048$ [22]) costs 84.4 μs, digital signature (i.e. Boneh-Lynn-Shacham (BLS) signature [26]) generation 43.5 μs and verification 136.1 μs. Based on the electrical grid size in the UK [27], we set $\mathrm{N}_{\mathrm{d}} = 14$ (N_{d} - number of DNOs in the grid), $\mathrm{N}_{\mathrm{wg}} = 140$ (N_{wg} - number of WGs in the grid), $\mathrm{N}_{\mathrm{ng}} = 32$ (N_{ng} - number of NGs connected to each WG), $\mathrm{N}_{\mathrm{bg}} = 28$ (N_{bg} - number of BGs connected to each NG), and vary N_{sm} from 100 to 420 (N_{sm} - number of SMs connected to each BG) and N_{s} from 1 to 33 (N_{s} - number of suppliers in the electricity market). Fig. 8 depicts the variation of computational costs at different entities vs. N_{sm} and N_{s}. Adding more SMs to the grid will affect (increase linearly) only the computational costs at BGs. Introducing new suppliers to the electricity market will only affect (increase slightly) the computational cost at DCC. Thus, our protocol, in contrast to other work [17], [18], scales well in terms of SMs and suppliers. It is also worth noting that each SM generates only one report in each timeslot, but still the consumption data included in the report is delivered (in an aggregated form) to three different entities: SM's regional

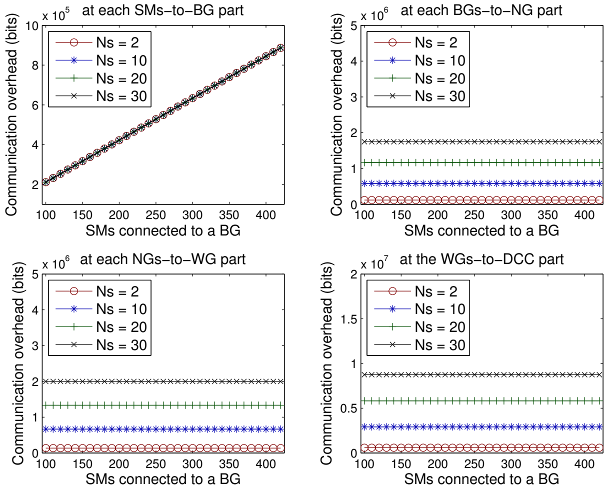

Fig. 9. Communication overhead of the aggregated data reporting protocol.

DNO, the SM user's supplier and the TSO. Thus, our protocol is cost efficient in terms of computational costs at SMs too.

We also ran simulations on Matlab with $|C_i|$ and $|ID|$ 2,048-bit and 32-bit long, respectively. Fig. 9 depicts the variation of communication overheads vs. N_{sm} and N_s. Similarly, adding more SMs will only affect (increase linearly) the communication overheads at the SMs-to-BG part of the grid. An increase in N_s will lead to a linear increase of the communication overhead for the grid excluding the SMs-to-BG part.

B. Attributable Meter Reading Reporting

During a protocol execution, SM performs only two computationally expensive operations, i.e. one signature verification and one asymmetric decryption. Also, as a supplier can request data as many times as it needs, there is no need for the supplier to return any acknowledgements for the data received. This can further reduce communication overheads between suppliers and SMs as well as computational costs imposed on SMs.

C. SM-to-IHD Communication

The costs imposed on SMs and IHDs are minimal, as data handled by them are secured using computationally 'inexpensive' crypto methods, i.e. symmetric encryptions and HMACs.

VII. Conclusion

This paper has described the design of MUSP, a secure and user-privacy-preserving system for smart metering. The novelty of MUSP lies in its support to the following services:

- aggregated data reporting to operators and suppliers,
- MRs reporting to users,
- attributable MRs reporting to suppliers,
- attributable MRs release frequency adjustment by users,
- attributable MRs release on demand, and
- easy supplier switching

in an efficient, scalable, secure and privacy-preserving manner. Also, the MUSP system is readily applicable to liberalized electricity markets, such as the UK electricity market.

Acknowledgment

This research is supported by the Engineering and Physical Sciences Research Council (EPSRC) and Toshiba Research Europe Limited under Grant [EP/I501541/1].

References

[1] H. Farhangi. The path of the smart grid. *Power and Energy Magazine, IEEE*, 8(1):18–28, Jan.-Feb. 2010.

[2] Z.M. Fadlullah, M.M. Fouda, N. Kato, X. Shen, and Y. Nozaki. An early warning system against malicious activities for smart grid communications. *Network, IEEE*, 25(5):50–55, Sept. 2011.

[3] U.S. NIST, Guidelines for smart grid cyber security (vol. 1 to 3), NIST IR-7628, Aug. 2010.

[4] P. McDaniel and S. McLaughlin. Security and privacy challenges in the smart grid. *Security Privacy, IEEE*, 7(3):75 –77, May 2009.

[5] H. Khurana, M. Hadley, Ning Lu, and D.A. Frincke. Smart-grid security issues. *Security Privacy, IEEE*, 8(1):81 –85, Jan.-Feb. 2010.

[6] Zhong Fan, P. Kulkarni, S. Gormus, C. Efthymiou, G. Kalogridis, M. Sooriyabandara, Z. Zhu, S. Lambotharan, and W.H. Chin. Smart grid communications: Overview of research challenges, solutions, and standardization activities. *Comm. Surveys Tutorials, IEEE*, 15(1), 2013.

[7] Jing Liu, Yang Xiao, Shuhui Li, Wei Liang, and C. L. Philip Chen. Cyber security and privacy issues in smart grids. *Communications Surveys Tutorials, IEEE*, 14(4):981–997, 2012.

[8] G. Kalogridis, M. Sooriyabandara, Z. Fan, and M.A. Mustafa. Toward unified security and privacy protection for smart meter networks. *Systems Journal, IEEE*, 8(2):641–654, June 2014.

[9] E. L. Quinn. Privacy and the new energy infrastructure. *Social Sience Research Networks (SSRN)*, Feb. 2009.

[10] C. Efthymiou and G. Kalogridis. Smart grid privacy via anonymization of smart metering data. In *Smart Grid Communications (SmartGridComm), 2010 First IEEE Int. Conf.*, pages 238–243, Oct. 2010.

[11] Hsiao-Ying Lin, Shiuan-Tzuo Shen, and B.P. Lin. A privacy preserving smart metering system supporting multiple time granularities. In *Software Security and Reliability Companion, IEEE Int. Conf. on*, 2012.

[12] F.G. Mármol, C. Sorge, O. Ugus, and G.M. Pérez. Do not snoop my habits: preserving privacy in the smart grid. *Communications Magazine, IEEE*, 50(5):166–172, 2012.

[13] Fengjun Li, Bo Luo, and Peng Liu. Secure information aggregation for smart grids using homomorphic encryption. In *Smart Grid Communications, IEEE Int. Conf. on*, pages 327 –332, Oct. 2010.

[14] Flavio D. Garcia and Bart Jacobs. Privacy-friendly energy-metering via homomorphic encryption. In *Security and Trust Management*, volume 6710 of *Lecture Notes*, pages 226–238. Springer Heidelberg, 2011.

[15] Pan Deng and Liuqing Yang. A secure and privacy-preserving communication scheme for advanced metering infrastructure. In *Innovative Smart Grid Technologies (ISGT), 2012 IEEE PES*, Jan. 2012.

[16] Fengjun Li and Bo Luo. Preserving data integrity for smart grid data aggregation. In *SmartGridComm, IEEE*, pages 366–371, 2012.

[17] R. Lu, X. Liang, X. Li, X. Lin, and X. Shen. Eppa: An efficient and privacy-preserving aggregation scheme for secure smart grid communications. *Parallel and Distrib Syst, IEEE Trans*, 23(9), 2012.

[18] Sushmita Ruj and Amiya Nayak. A decentralized security framework for data aggregation and access control in smart grids. *Smart Grid, IEEE Transactions on*, 4(1):196–205, 2013.

[19] C. Rottondi, G. Verticale, and C. Krauss. Privacy-preserving smart metering with multiple data consumers. *Computer Networks*, 57, 2013.

[20] C. Rottondi, G. Verticale, and C. Krauss. Distributed privacy-preserving aggregation of metering data in smart grids. *Selected Areas in Communications, IEEE Journal on*, 31(7):1342–1354, July 2013.

[21] M.A. Mustafa, Ning Zhang, G. Kalogridis, and Zhong Fan. Desa: A decentralized, efficient and selective aggregation scheme in ami. In *Innovative Smart Grid Technologies (ISGT), 2014 IEEE PES*, 2014.

[22] P. Paillier. Public-key cryptosystems based on composite degree residuosity classes. In *Advances in Cryptology*, pages 223–238, 1999.

[23] C. Fontaine and F. Galand. A survey of homomorphic encryption for nonspecialists. *EURASIP Journal on Information Security*, Jan 2007.

[24] Pbc library, http://crypto.stanford.edu/pbc/.

[25] Miracl library, http://certivox.org/display/ext/miracl.

[26] D. Boneh, B. Lynn, and H. Shacham. Short signatures from the weil pairing. In *Advances in Cryptology*, volume 2248, pages 514–532, 2001.

[27] ofgem - electricity. Internet: www.ofgem.gov.uk/electricity [22.01.2015].