SECURITY AND COMMUNICATION NETWORKS
Security Comm. Networks 2014; **7**:234–244
Published online 18 January 2013 in Wiley Online Library (wileyonlinelibrary.com). DOI: 10.1002/sec.699

SPECIAL ISSUE PAPER

SESA: an efficient searchable encryption scheme for auction in emerging smart grid marketing†

Mi Wen[1,2]*, Rongxing Lu[2], Jingsheng Lei[1], Hongwei Li[2,3], Xiaoghui Liang[2] and Xuemin (Sherman) Shen[2]

[1] School of Computer Engineering, Shanghai University of Electric Power, Shanghai 200090, China
[2] Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, Ontario N2L 3G5, Canada
[3] School of Computer Science and Engineering, University of Electronic Science and Technology of China, Chengdu, China

ABSTRACT

Distributed energy resources (DERs), which are characterized by small-scale power generation technologies to provide an enhancement of the traditional power system, have been strongly encouraged to be integrated into the smart grid, and numerous trading strategies have recently been proposed to support the energy auction in the emerging smart grid marketing. However, few of them consider the security aspects of energy trading, such as privacy preservation, bid integrity, and prefiltering ability. In this paper, we propose an efficient searchable encryption scheme for auction (SESA) in emerging smart grid marketing. Specifically, SESA uses a public key encryption with keyword search technique to enable the energy sellers (e.g., DERs) to inquire suitable bids while preserving the privacy of the energy buyers. Additionally, to facilitate the seller to search for detailed information of the bids, we also propose an extension of SESA to support conjunctive keywords search. Security analysis demonstrates that the proposed SESA and its extension can achieve data and keyword privacy, bid integrity and trapdoor unforgeability. Simulation results also show that both SESA and its extension have less computation and communication overhead than the existing searchable encryption approaches. Copyright © 2013 John Wiley & Sons, Ltd.

KEYWORDS
smart grid; privacy preserving; keyword search; energy auction

***Correspondence**
Mi Wen, School of Computer Engineering, Shanghai University of Electric Power, Shanghai 200090, China.
E-mail: mi.wen@uwaterloo.ca

†Please ensure that you use the most up-to-date class file, available from the SEC Home Page at http://www3.interscience.wiley.com/journal/114299116/home.

1. INTRODUCTION

Growing demand for electricity, upcoming fossil-fuel shortage, and CO_2 emission crises have recently invoked an urgent need in incorporating renewable energy sources into the power grid. Such a trend is commonly known as distributed generation (DG) [1]. In the trend of DG, distributed energy resources (DERs) have been encouraged to participate in energy marketing to facilitate competition among different energy providers. However, how to negotiate with different energy providers and energy consumers is a challenging issue in DG [2]. In order to address this challenge, smart grid, which is composed of many entities (intelligent electricity distribution devices, advanced sensors, two-way automated metering infrastructure, and specialized computer systems to enhance the operation performance [2]), has received significant attention in recent years. Smart grid can accelerate the integration of distributed energy suppliers, DERs, and microgrids [3], and thus, it potentially makes power generation, transmission, and distribution become the next big e-business operating mostly under autonomous control [4]. As a result, smart grid has been recognized as an emerging electricity market.

In this paper, we consider an energy auction scenario, where DERs and energy consumers respectively act as sellers and buyers in smart grid marketing. The energy trading process between sellers and buyers includes the following steps: each DER publishes a document about its energy description (e.g., starting price, type, mount, announcement date, and expire date). When a buyer wants to buy some energy, he or she first puts his or her bid on an auction server (AS). The AS manages an auction on the basis of the bids and asks from buyers and sellers, respectively. At the end of the auction, DERs decide the winner of the auction by judging the bids on the AS according to some certain selection criterion.

The bids coming from various DERs will refer to the actual market prices. In addition, it should be competitive to the actual market prices. The information from EBs is important and sensitive not only for auction in smart grid marketing but also for market analysis and decision support, predicting future demand and prices, and monitoring in case of unexpected behavior. Conventionally, in an auction of the energy marketing, bids are calculated without considering the security and privacy issues of the DERs and EBs. If one of the bidders wants to win in the auction process, it may maliciously modify the bids from other bidders. These malicious behaviors will lead to unfair auction and be harmful to grid stability and power quality. Furthermore, if the AS is totally trusted and if all the bids are opened and winners are decided by it, the AS will be a vulnerability point of the whole smart grid energy marketing. Thus, another server is needed to help in selecting the bids and winners from the AS. Because of the fact that there may be large amount of bids in the smart grid energy marketing, how to search the bids by a keyword (or some keywords) from AS without exposing the real value of the bids is a new challenge in emerging smart grid energy marketing.

In order to protect users' information privacy and security during the auction process, each buyer should protect their bidding information and let it not be known by the unauthorized users, including the AS. While at the same time, it enables the sellers to query the AS about the demanded bids. Although many auction models (e.g., [2,10,12,13]) were established respectively for smart grid energy marketing, few of them take the privacy or security of the DERs into consideration. Recently, various security vulnerabilities and threats have been studied in the research literatures [11,17,18]. Lu *et al.* [5] used a super-increasing sequence to structure multi-dimensional data and encrypt the structured data by the homomorphic Paillier cryptosystem technique. Li *et al.* [6] proposed an efficient demand response scheme to achieve privacy preserving demand aggregation and efficient response. However, because these encryption schemes cannot be searched, they are not suitable for auction in smart grid marketing. On the other hand, some of the traditional auction schemes [7,8] can achieve bidding privacy, but they cannot support keyword search or bids filtering.

In this paper, we propose an efficient searchable encryption scheme for auction (SESA) in smart grid marketing. This scheme considers both the public key-based encryption and keyword search techniques. It can achieve privacy preservation, searchable ability, and bids filtering, as well as other security features including confidentiality, authenticity, and integrity. The contributions of this paper are twofold.

(1) Firstly, we propose a novel SESA to achieve searchable encryption, by modifying the proxy re-encryption with keyword search scheme [9]. The security analysis demonstrates that SESA can achieve confidentiality, data and keyword privacy, authenticity, and data integrity.
(2) Secondly, we construct an extended version of SESA to support conjunctive keywords search. It enables the user to question the AS more flexibly.

The remainder of this paper is organized as follows. In Section 2, we introduce related works. In Section 3, we recall some preliminary knowledge. Then, we describe the smart grid marketing architecture, security requirement, and design goal in Section 4. Next, we present the proposed SESA and its extension in Sections 5 and 6, followed by its security analysis and performance evaluation in Sections 7 and 8, respectively. Finally, we conclude the paper in Section 9.

2. RELATED WORK

The traditional auction schemes can be divided into two categories: open outcry and sealed bids. Open outcry can further be separated into English auctions and Dutch auctions [14]. In English auctions, the value of the bid is public, and the price of the bid must be higher than the current price. The highest bidder is the winner at the end of the bidding phase. There are many famous English auction web sites (e.g., Yahoo! and eBay) [8]. The Dutch auction is almost the same as the English auction, except that it begins with the top price. In a sealed bid auction, the bidders write the price and quantity of their bid on a sheet of paper, and then, they seal the sheet and give it to the auctioneer. The auctioneer collects all the sealed sheets and opens them after the deadline to determine the winner. A sealed bid auction can be separated into two kinds, first-price sealed bid and second-price sealed bid.

The bidding manner has been extensively studied and various bidding models are presented in the power market [10,12]. Among the various methods, the simplest way is to estimate the market clearing price of the next time and then present the bid with a lower price than the estimated one. The second method is to estimate the behaviors of the rivals and to present the bid [12]. The third method is based on the game theory [13] with oligopolistic strategy such as Cournot model and supply function models [10]. But, few of them consider the privacy of the bidders and the energy providers. In electronic auction systems, Chang [7] and Li [8] both present anonymous auction protocol with freewheeling bids. However, bidding privacy cannot be achieved in [7], and both of them cannot support keyword search or any other filtering.

The concept of public key encryption with keyword search (PEKS) was proposed by Boneh *et al.* [16], which supports the keyword search on encrypted data. Other schemes focusing on constructing keyword encryption were extensively discussed, such as [19]. The Public Key Encryption with Conjunctive-Subset Keywords Search

(PECSK) scheme [20] supports conjunctive-subset keywords search. But it is only a keyword search scheme. In [15], Liu *et al*. presented an efficient privacy preserving keyword search (EPPKS) scheme in cloud computing. It is one of the few schemes that integrate both the message encryption and keyword search properties. However, when the server finds a tag matching the trapdoor in EPPKS [15], the server has to compute an intermediate result to help the user to recover the message, which costs communication and computation overhead.

3. PRELIMINARIES

In this section, we will briefly describe the basic definition and properties of bilinear pairings and PEKS.

3.1. Bilinear pairing

Bilinear pairing is an important cryptographic primitive [21]. Let G_1 and G_2 be two cyclic multiplication groups of prime order q. Let a and b be elements of Z_q^*. We assume that the discrete logarithm problem in both G_1 and G_2 are hard. g is a generator of G_1. A bilinear pairing is a map $e: G_1 \times G_1 \rightarrow G_2$ with the following properties:

(1) Bilinear: $e(g^a,h^b) = e(g,h)^{ab}$ for any $(g,h) \in G_1^2$.
(2) Non-degenerate: $e(g,h) \neq 1_{G_2}$ whenever $g,h \neq 1_{G_1}$.
(3) Computable: There is an efficient algorithm to compute $e(g,h) \in G_2$ for all $(g,h) \in G_1^2$.

Definition 1. *A bilinear parameter generator $\mathcal{G}en$ is a probabilistic algorithm that takes a security parameter κ as input, and outputs a five-tuple (q,P,G_1,G_2,e).*

3.2. Public key encryption with keyword search

Formally, PEKS studies the problem of searching on data that is encrypted using a public key system. In this setting, the server plays the role of data warehouse for the receiver. The framework of PEKS scheme [20] is illustrated in Figure 1. With the PEKS scheme, the workflow of the underlying application consists of two phases.

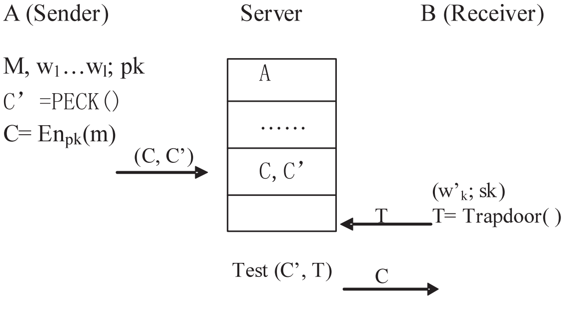

Figure 1. A PECSK scheme framework.

(1) In the first phase, a sender encrypts its message, runs PEKS() to generate some keyword tags for the message, and stores the ciphertext and the tags at the server.
(2) In the second phase, the receiver runs Trapdoor() to generate a trapdoor for each selected keyword and sends the trapdoors to the server, which will run Test() to search over the tags attached to each encrypted message.

Most of the existing literatures focus on the part of keyword search techniques PEKS(), they assume that any public key encryption can be used as En(). It does work in real applications, but if the foundation of En() and PEKS() are different, the overhead may be large. If there is a scheme that can combine the public key encryption En() and keyword search technique PEKS() together, it may improve the performance of the system. As such, it can be a real searchable encryption scheme.

4. SYSTEM MODEL

In this section, we formalize the system model, security requirements, and identify our design goals.

4.1. Smart grid marketing architecture

Smart grid marketing refers to a system that enables small producers to generate and sell electricity at the local level. As shown in Figure 2, there are energy sellers (e.g., DERs), energy buyers (EBs), and auction managers. The auction managers are two servers: a registration server (RS) and an AS.

RS In energy marketing, an RS is used to initiate the system at the beginning of the auction and when the bidding is finished, it will select the winner according to the criteria of the DERs. The RS is trustworthy, and it will send some keywords from the DERs to the AS to search for their wanted bids. The winner may be selected from these pre-filtered bids.

AS AS is used in a continuous sealed-bid auction in which traders submit offers to buy (bid) or offers to sell (ask) at any time during the trading period. The AS is semi-trusted, and it cannot know the content of the EBs' bids, but it can test if the message has tags such as the seller's query.

DER DERs can open the bids by themselves. However, because the number of distributed bids from EBs may be large, to improve the efficiency, the RS will act as a proxy for the DERs to select the winners.

EB Energy is bought from or sold to the grid depending on the availability, demand, and price of energy. Each EB will send its sealed bid to the AS. Because of the large amount of buyers, the bids may be conducted with the competition of others.

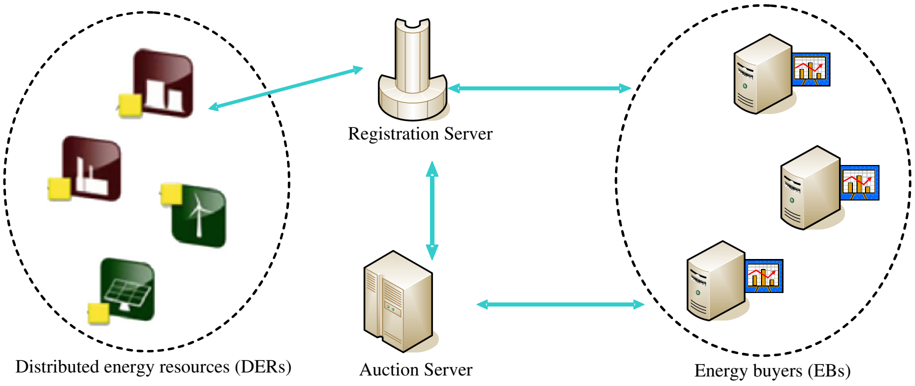

Figure 2. Smart grid marketing architecture.

4.2. Security requirements

We assume that the communication between EBs and server is untrustworthy. That is, various adversaries such as eavesdroppers and tampers may be present. If a large number of EBs are competitive to buy a certain type of energy from DERs, it is reasonable to enable the RS to query the AS and select one or a group of winners according to the criteria of the DERs.

We define the security requirements for our SESA and will show the fulfillment of these requirements after presenting the design details.

- *Data privacy*: The data owner can resort to the public key cryptography to encrypt the data before outsourcing and successfully prevent the unauthorized entities, including the AS, from prying into the outsourced data.
- *Bid integrity*: The bids information and queries should not be changed by the malicious users or the illegal competitors; that is, if the competitor $\mathcal{A}$ maliciously modified the price or other information of EB_i, it may lead to a situation where EB_i cannot be selected by the RS.
- *Keyword privacy*: As users usually prefer to keep their search from being exposed to others, including the AS, the most important concern is to hide what in their bids and what the RS is inquiring, that is, the keywords indicated by the corresponding trapdoor. Thus, the trapdoor should be generated in a cryptographic way to protect the query keywords.
- *Trapdoor unforgeability*: DER generates his or her trapdoor information on the basis of his or her keyword and secret key. After receiving the trapdoor, the AS can test this trapdoor with keyword tags. The most important thing is that others (including the AS) can achieve nothing from the trapdoor; that is, the AS cannot forge a new trapdoor on the basis of the old ones.

4.3. Design goals

To enable searchable encryption for effective utilization of outsourced energy bids under the aforementioned model, our design goal is to develop a SESA in emerging smart grid marketing and achieve the security of the bids and efficient keyword search as follows.

- The proposed scheme should achieve security as mentioned in the security requirements, that is, the data privacy, keyword privacy, data integrity, and trapdoor unforgeability.
- The proposed scheme should achieve both one keyword and conjunctive keywords search.
- The proposed scheme should achieve the communication and computation efficiency, compared with other searchable encryption schemes.

5. PROPOSED SESA

In this section, we propose an efficient SESA in emerging smart grid marketing, which mainly consists of the following four phases: registration phase, bidding phase, pre-filtering phase, and decision-of-winner phase. For our auction system, we assume that there is a local RS that can bootstrap the system. Specifically, in this system initialization phase, given the security parameter 1^k, RS first generates (q,g,G_1, G_2,e) by running $\mathcal{G}en\left(1^k\right)$, where q is a k-bit prime number. Let G_1 and G_2 be two cyclic multiplication groups. $Sig(\mathcal{G}, U, V)$ is an ID-based signature scheme [24]. Furthermore, we will need three hash functions $H_1: \{0,1\}^* \rightarrow G_1$, $H_2: \{0,1\}^* \rightarrow G_1$, $H_3: G_2 \rightarrow \{0,1\}^*$. RS publishes the system parameters as (q,g,G_1,G_2,e,H_1,H_2,H_3).

5.1. Registration phase

In order to maintain security of the network against attacks and the fairness among customers and providers, the local RS may control the access of each DER and EB. The energy marketing announces two prices: the price for selling energy and the price for buying energy in the smart grid marketing. The DERs adjust their bidding price after negotiating with the other units on the basis of the grid prices, considering their operational cost and local demands.

In our scheme, there are n DERs and m EBs in the energy marketing. For each DER_i $(i = 1,\ldots,n)$ and EB_j $(j = 1,\ldots,m)$, when they register, the RS picks one random numbers be: $x_i \in Z_q^*$ and sets $pd_i = g^{x_i}$, (pd_i, x_i) is $DER_i^{'}$ s public/private key pair. For each EB_j $(j = 1,\ldots,m)$, the RS randomly chooses a master key $s \in Z_q^*$ and assigns an ID-based key pair $\left(H_1\left(ID_{EB_j}\right), H_1^s\left(ID_{EB_j}\right)\right)$ to DER_i for signature and denotes it as (vk_j, ssk_j).

In the energy marketing, the DER_i will publish its energy information $m_i = (p_i, GID_i, Ts, Lo_i, Am_i, T_N)$ publicly, where p_i is the initial price, GID_i is the identification of the energy, Ts is the timestamp, Lo_i is the energy resource location, Am_i is the amount of the energy, and T_N is the unique serial number of the deposit energy information. The RS will store the information from each DER_i as a tuple (DER_i, m_i) in its database. Also, EB_j will register its personal information $e_j = (Lo_j, Rep_j, Ty_j, \triangle)$ on the RS, where Lo_j is its location, Rep_j is its reputation about its history trades (which also will be verified by the RS, but it is not our paper's focus), Ty_j is the demanded energy types, and Δ is the other information of EB_j. The RS also stores the information from each EB_j as a tuple (EB_j, e_i) in its database.

5.2. Bidding phase

In order to achieve the nearly real-time energy bidding, each EB_j will choose its interested energy to bid. The bidding is performed as following steps.

(1) EB_j gets an ID-based signature key pair as (vk_j, ssk_j) from RS. The public key is represented as $A = vk_j$, and the private key ssk_j is kept secretly.
(2) EB_j selects a random $r_j \in Z_q^*$ and generates a $bid_j = (EB_j, pr_j, GID_i, Cb_j, Rep_j)$, where pr_j is the price of the bid, Cb_j is the amount of the energy that EB_j wants to buy, Rep_j is $EB_j^{'}$ s reputation. Then, EB_j computes $C_j = H_3(e(g, H_2(A)^{r_j})) \oplus bid_j$.
(3) In order to maximize the probability of winning in the auction, EB_j selects a keyword w_j to represent his or her bid (e.g., the reputation or required amount). Next, EB_j computes a tag on the keyword as $t_j = e\left(g, H_1\left(w_j\right)^{r_j}\right)$. Then, he or she computes $B_j = \left(g^{x_i}\right)^{r_j}$ and $F_j = H_3(t_j)$. He or she outputs $C'_j = \left(B_j, F_j\right)$.
(4) EB_j generates a signature $S_j = Sig_{ssk_j}\left(C_j, C'_j\right)$. $\left(C_j, C'_j\right)$ is the signed message.
(5) EB_j sends the encrypted message $K_j = (GID_i, A, C\ j, C'_j, S_j)$ to the AS.
(6) The AS stores this information from EB_j as a tuple (EB_j, K_j) in its bid table.

5.3. Pre-filtering phase

The goal of bids pre-filtering is to quickly identify potential winner or winners from all the bids in the AS's bid table.

For example, if DER_i wants to filter the bids for energy GID_i according to the user's reputation $w_i^{'}$, DER_i generates a trapdoor $t_{w_i^{'}}$ in advance and sends it to the RS. In order to preserve the privacy of DER_i and EB_j, the trapdoor $t_{w_i^{'}} = H_1(w_i^{'})^{1/(x_i)}$ is a ciphertext of the value $w_i^{'}$. Then, RS will send $t_{w_i^{'}}$ to the AS. On receiving the message from RS, for each bid in the AS's bid table, the AS will test if the given $C_j^{'}$ satisfies the selection criterion $t_{w_i^{'}}$ of DER_i:

(1) Message verification:
   (a) The AS verifies signature S_j on message $\left(C_j, , C'_j\right)$ with respect to the public key A.
   (b) If it fails, the AS will reject this bid; else the AS will go on testing.
(2) Trapdoor and tag test:

The AS tests if $H_3\left(e\left(B_j, t_{w_i^{'}}\right)\right) = F_j$. If so, which means $w_j = w'_i$, the encrypted bid C_j will be stored in a filtered array **W[]**. Later, **W[]** will be transferred to the RS. If not, AS will go on testing the other bids. The correctness of $H_3\left(e\left(B_j, t_{w_i^{'}}\right)\right) = F_j$ is as follows:

$$\begin{aligned} & H_3\left(e\left(B_j, t_{w'_i}\right)\right) \\ & = H_3\left(e\left(\left(g^{x_i}\right)^{r_j}, H_1\left(w'_i\right)^{1/(x_i)}\right)\right) \\ & = H_3\left(e\left(g, H_1\left(w'_i\right)^{r_j}\right)\right) \\ & = F_j = H_3\left(e\left(g, H_1\left(w_j\right)^{r_j}\right)\right) \end{aligned} \tag{1}$$

5.4. Decision-of-winner phase

On receiving filtered bids array **W**[] from the AS, the RS can decrypt each C_j in **W**[] as $bid_j = C_j \oplus H_3\left(e\left(B_j, H_2(A)\right)^{1/x_i}\right)$ by using DER'_i secret key x_i; otherwise, C_j will be discarded. The correctness of the decryption is shown as follows:

$$\begin{aligned} & C_j \oplus H_3\left(e\left(B_j, H_2(A)\right)^{1/x_i}\right) \\ & = H_3(e(g, H_2(A)^{r_j})) \oplus bid_j \oplus H_3\left(e((g^{x_i})^{r_j}, H_2(A))^{1/x_i}\right) \\ & = H_3(e(g, H_2(A)^{r_j})) \oplus bid_j \oplus H_3(e(g, H_2(A)^{r_j})) \\ & = bid_j \end{aligned} \tag{2}$$

We assume that there are t decrypted bids, and the bids will be put in a sorted array list **B**[] according to their price in a descending order. Because of the special difficulties in energy storage and profit maximization of the auction in nature, the winner-selection criterion from DER_i should achieve two goals: one is that the total sales should be as high as possible; the other is that the sum of the demanded amount of the winners should be as close to the available energy demand Am_i as possible. The selected winners will be stored in an array list **S**[] by using Algorithm 1. Finally, the RS will secretly deliver the winners list **S**[] to DER_i.

DOI: 10.1002/sec

6. EXTENDED SESA WITH CONJUNCTIVE KEYWORDS SEARCH

The SESA can be extended to support conjunctive keywords search, with which the DERs can get more detailed information about the bids. Because the decision-of-winner phase in this extension is same as that in SESA, we only introduce the registration phase, bidding phase, and pre-filtering phase as follows.

6.1. Registration phase

In this extended scheme, there are also n DERs and m EBs in the energy marketing. For each DER_i $(i = 1,\ldots,n)$ and EB_j $(j = 1,\ldots,m)$, when they register, the RS picks two random numbers $x_i \in Z_q^*$ and sets $pd_i = g^{x_i}$. (pd_i, x_i) is DER_i's public/private key pair. The RS randomly chooses a master key $s \in Z_q^*$ and assigns an ID-based key pair $\left(H_1\left(ID_{EB_j}\right), H_1^s\left(ID_{EB_j}\right)\right)$ for each EB_j $(j = 1,\ldots,m)$. The key pair is represented as (vk_j, ssk_j). Similar to the SESA, the DER_i will publish its energy information $m_i = (p_i, GID_i, Ts, Lo_i, Am_i, T_N)$ publicly. The RS will store the information from each DER_i as a tuple (DER_i, m_i) in its database. Also, EB_j will register its personal information $e_j = (Lo_j, Rep_j, Ty_j, \triangle)$ on the RS. The RS also stores the information from each EB_j as a tuple (EB_j, e_i) in its database.

Algorithm 1. Winner Selection(**B**,**S**).

```
Input: B[ ]
Output: S[ ]
$c \leftarrow Am_i$,c is the remain energy amount;
$k1, k2 \leftarrow 0$; $k = t$; S[ ] $\leftarrow \phi$.
If two bids have the same price, the one requires
bigger amount will be first served.
while ($k! = 0$) do
    for each B[$k1$] do
        if (B[$k1$].$price$ = B[$k1 + 1$].$price$) and
        (B[$k1$].$amount$ < B[$k1 + 1$].$amount$) then
            $temp \leftarrow$ B[$k1$]; B[$k1$] $\leftarrow$ B[$k1 + 1$];
            B[$k1 + 1$] $\leftarrow temp$;
        end
        $k1 + +$;
    end
    $k - -$;
end
$k1 \leftarrow 0$;
for each B[$k1$] do
    if (B[$k1$].$amount < c$) then
        S[$k2$] $\leftarrow$ B[$k1$], $k2 + +$;
        $c \leftarrow c -$ B[$k1$].$amount$;
    end
    $k1 + +$;
end
```

In order to provide more convenience for the DERs to have detailed filtering, that is, let them achieve the conjunctive keywords search from the AS, each EB_j will select a keywords set $W_j = \{w_{j1}, w_{j2},\ldots,w_{jL}\}$ to characterize his or her bid. Without loss of generality, the location of each type of keyword in the keywords set $W_j = \{w_{j1}, w_{j2},\ldots,w_{jL}\}$ is fixed. For instance, w_1 denotes the type of the source address keyword, w_2 denotes the type of energy amount keyword, and so on. Keywords in the DER'_i tag and EB'_j trapdoor are in the same order.

6.2. Information encryption

Each EB_j publishes its bid as the following steps:

(1) EB_j gets an ID-based signature key pair as (vk_j, ssk_j). The public key is denoted as $A = vk_j$, and the private key ssk_j is kept secretly.
(2) EB_j selects a random number $r_j \in Z_q^*$ and generates a $bid_j = (EB_j, pr_j, GID_i, Cb_j, Ts_j, \triangle)$, where pr_j is the price of the bid, Cb_j is the amount of the energy that EB_j want to buy, and Δ is the other information of EB_j. Then, EB_j computes $C_j = H_3(e(g, H_2(A)^{r_j})) \oplus bid_j$.
(3) EB_j computes a tag for each keyword as $t_{jk} = e(g, H_1\ (w_{jk})^{r_j}), (k = 1,\ldots,L)$, $B_j = (g^{x_i})^{r_j}$. EB_j outputs $C'_j = \left(B_j, t_{jk}(k = 1,\ldots,L)\right)$.
(4) EB_j generates a signature $S_j = S_{ssk}\left(C_j, , C'_j\right)$, where the message to be signed is the tuple $\left(C_j, C'_j\right)$.
(5) EB_j sends the encrypted messages $K_j = \left(AC_j, S_j, C'_j\right)$ to the AS.
(6) The AS will store this information from EB_j as a tuple (EB_j, K_j) in its bid table.

6.3. Pre-filtering phase

If the DER_i needs to filter the bids by using some criteria (e.g., reputation and location). It will generate a keywords set $Q_i = \{w_{E1}, w_{E2},\ldots,w_{Et}\}$. Then, DER_i generates a trapdoor t_{Q_i} and sends it to the RS. At the end of the auction, the RS will transfer this trapdoor t_{Q_i} to the AS to filter the bids. Without loss of generality, we assume that $\{E1, E2,\ldots,Et\}$ is the subset of $\{j1, j2,\ldots,jL\}$.

(1) DER_i generates a trapdoor on the keywords Q_i as $t_{Q_i} = (H_1(w_{E1}).H_1(w_{E2})\ldots H_1(w_{Et}))^{1/(x_i)}$. DER_i sends $(t_{Q_i}, \{E1, E2,\ldots,Et\})$ to the RS. The RS transfers them to the AS.
(2) For each C_j in GID_j.s bid table, the AS will test if C'_j satisfies EB_j's reqirement:
   (a) Message verification:
      (i) The AS verifies signature S_j on message $\left(C_j, C'_j\right)$ with respect to the public key A.
      (ii) If it fails, the AS will reject this bid; else the AS will go on testing.

(b) The AS tests if $H_3\left(e\left(B_j,,t_{Q_i}\right)\right) = H_3\left(\prod_{v=E1}^{Et} t_v\right)$. If so, C_j will be stored in an array list W[]; if not, C_j will be rejected. The correctness of the test is shown as follows:

$$\begin{aligned}
&H_3\left(e\left(B_j, t_{Q_i}\right)\right)\\
&= H_3\left(e\left(\left(g^{x_i}\right)^{r_j}, \left(H_1(w_{E1}).H_1(w_{E2})\dots H_1(w_{Ek})\right)^{1/(x_i)}\right)\right)\\
&= H_3\left(e\left(g, H_1(w_{E1})^{r_j}\right).e\left(g, H_1(w_{E2})^{r_j}\right), \dots e\left(g, H_1(w_{Ek})^{r_j}\right)\right)\\
&= H_3\left(\prod_{v=E1}^{Ek} t_v\right)
\end{aligned} \tag{3}$$

7. SECURITY ANALYSIS

In this subsection, we analyze the security properties of the proposed SESA. In particular, following the security requirements discussed earlier, our analysis will focus on how the proposed SESA can achieve the goals. The extension can also achieve these properties.

- *The individual EB's bid is privacy preserving in the proposed SESA*: In the proposed SESA, EB's bidding information is encrypted by its secret number r_j as $C_j = H_3(e(g, H_2(A)^{r_j}))\oplus bid_j$. Anyone, including the AS, who does not know the secret number r_j cannot recover bid_j from the ciphertext C_j. Thus, if a bidder does not win the auction, in the proposed SESA, nobody can have any information about the bidder from its bid.
- *The authentication and data integrity of the individual EB's bid is achieved in the proposed SESA*: In SESA, each EB's bidding information is signed by the ID-based signature scheme [24]. Because the ID-based signature $S_j = S_{ssk}\left(C_j, C'_j\right)$ is provably secure, the source authentication and data integrity can be guaranteed. As a result, adversary $\mathcal{A}$'s malicious behaviors in the smart grid communications can be detected in the proposed SESA.
- *The EB's keyword privacy and DER's trapdoor privacy are also achieved in the proposed SESA*: In the proposed SESA, on one hand, the keyword that EB chose to append on the encrypted bid is protected by a hash function. Anyone, including the AS, cannot recover w_j with the message C'_j. On the other hand, when RS delivers DER's query to the AS to search for certain type of bids, the query is also not delivered by plaintext; it is protected by a hash function. Thus, anyone who gets the trapdoor only knows the hash value of the keyword w'_i, and they do not know what the DER is really inquiring. Even when the AS does the verification of the tag and the trapdoor, it cannot know anything about the keyword except for whether they match or not.
- *The DER's trapdoor cannot be forged in the proposed SESA*: In the proposed SESA, although the AS can have lots of trapdoors from DERs, it cannot forge a valid new one from the existing old ones. That is, because all the keywords are blinded by a hash function, the AS cannot get the real value of the keywords.

It is illustrated in Table I that most of the auction schemes [12–14] for power market are lack of security concerns. While, in traditional electronic auction system, the work in [7] only achieves the confidentiality and data integrity, the work in [8] achieves confidentiality, data privacy, and data integrity. Only the proposed SESA can achieve additional keyword privacy and trapdoor unforgeability compared with [8].

Figure 3 shows that if the AS is compromised, the bids information and bidder's privacy will be disclosed in schemes [8,12–14]; only those in [8] and the proposed SESA can remain secure. But [8] cannot support keyword search on the bids, and there is only one winner in [8]; it is not applicable for energy auction in the smart grid. From the aforementioned analysis, we can see that the proposed SESA can provide enough security guarantees for auction in smart grid marketing.

8. PERFORMANCE ANALYSIS

8.1. SESA versus EPPKS

In this subsection, we will compare our SESA with the privacy preserving keyword search scheme (EPPKS) [15] in terms of the computation and communication overhead in the one keyword search process.

Computation: In our proposed SESA, the computation tasks include pairing operations and exponentiation operations, where the pairing operations are the most time-consuming tasks. Since the hash operation and number multiplication are too fast compared with the pairing operations, we will not take them into consideration in this subsection. For simplicity of description, the pairing operation and exponentiation operation are denoted as C_p and C_e, respectively.

For the proposed SESA, when EB_j generates an encrypted bid $\left(AC_j, S_j, C'_j\right)$, it requires three exponentiation operations and two pairing operations for bid encryption generation, i.e., $2C_\text{p} + 3C_\text{e}$. The DER_i or the RS needs one exponentiation operation to compute a trapdoor $t_{w'_i}$. After receiving the trapdoor from DER_i, the local AS needs to compute two pairings to verify the signature [24] and one pairing to test if there is a bid satisfying DER_i's query. Finally, DER_i or the RS requires one pairing operation and

Table I. Comparison of security properties.

Properties	[12–14]	[7]	[8]	SESA
Confidentiality	No	Yes	Yes	Yes
Data privacy	No	No	Yes	Yes
Bid integrity	No	Yes	Yes	Yes
Keyword privacy	No	No	No	Yes
Trapdoor unforgeability	No	No	No	Yes

SESA, searchable encryption scheme for auction.

DOI: 10.1002/sec

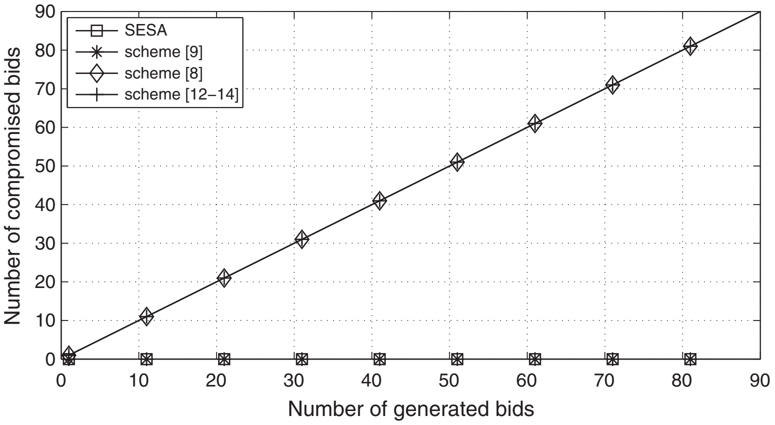

Figure 3. Fraction of compromised bids when the auction server is compromised. SESA, searchable encryption scheme for auction.

one exponentiation operation to decrypt the ciphertext if there are suitable bids.

In comparison, for EPPKS [15], it needs three pairing operations and six exponentiation operations to generate a data encryption on one keyword, i.e., $3C_p + 6C_e$. The seeker needs one exponentiation operation to compute a trapdoor T_{w_i}. And the server needs one pairing operation to test whether a given tag contains keyword T_{w_i}. Then, the server needs $2C_p + 2C_e$ more computation overhead to obtain an intermediate result of the partial decipherment. At last, it will cost the seeker C_e to recovery the ciphertext.

Table II indicates that SESA is more efficient than EPPKS [15]. Detailed experiments also are conducted on a Pentium IV 3-GHz system to study the execution time [22,23]. For G_1 over the Freeman-Scott-Teske (FST) curve, a single exponentiation operation in G_1 with 161 bits costs 1.1 ms, and the corresponding pairing operation costs 3.1 ms. The comparison of computation overhead is shown in Figure 4. We can see that SESA achieves totally lower execution times compared with EPPKS. Moreover, SESA can guarantee the integrity of the message, whereas EPPKS cannot achieve this property.

Table II. Comparison of computation complexity.

Computation	SESA	EPPKS
EB	$2C_p + 3C_e$	$3C_p + 6C_e$
AS	$3C_p$	$3C_p + 2C_e$
DER$_i$ or RS	$C_p + 2C_e$	$2C_e$

SESA, searchable encryption scheme for auction; EPPKS, efficient privacy preserving keyword search; EB, energy buyer; AS, auction server; DER, distributed energy resource; RS, registration server.

Communication: Most pairing-based cryptosystems need to work in a subgroup of the elliptic curve $E(F_q)$. By representing elliptic curve points using point compression [25], the length of the elements in G_1 and G_2 will be roughly 161 bits (using point compression) and 1024 bits, respectively. SHA-1 is used to compute the hash function, which yields a 160-bit output. Let the parameter n in EPPKS be 160 bits. The communications among the three entities of the proposed SESA can be divided into three parts: EB-to-AS, DER-to-AS, and AS-to-RS communications.

We first consider the EB-to-AS communication in SESA. In the information encryption phase, the data report is in the form of $K_j = \left(AC_j, S_j, C'_j\right)$. Because the length of ID-based signature [24] is two G_1 elements, the size of K_j

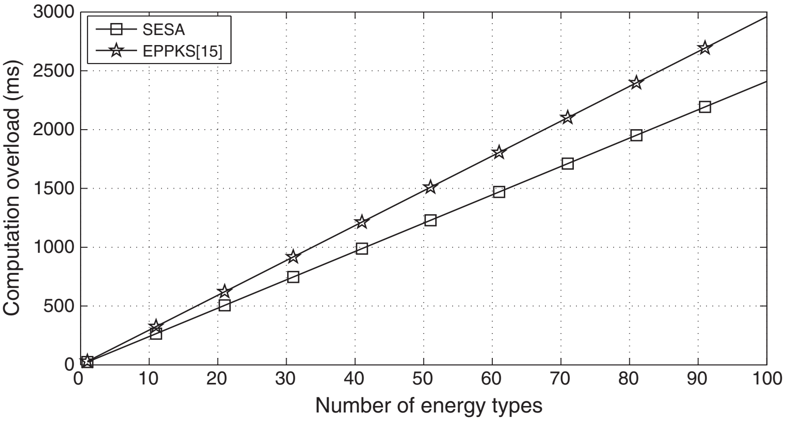

Figure 4. Comparison of computation overhead between searchable encryption scheme for auction (SESA) and efficient privacy preserving keyword search (EPPKS) schemes.

should be $160+160+161*2+160+161=963$ bits. In the DER-to-AS communication, DER needs to deliver a trapdoor t'_w to the AS, which is 160 bits, whereas in AS-to-RS communication, the AS will reply a ciphertext C_j to the EB if there is energy matching EB's demand, which is 160 bits.

In contrast, the user-to-server communication overhead in EPPKS is the message (C_m,C_w), which includes one G_1 element, two n-bit elements, and one hash element. The size is $161+2n+160=641$ bits. Then, the trapdoor T_{w_j} with the size of 160 bits will be sent from the user to the server. In the server-to-receiver communication, if there is a keyword match, the server will reply (C_m,C_ρ,C_w) to the receiver. Here, C_ρ is an element of G_2. The size of the reply is $161+160+2n+1024=1665$ bits. Table III and Figure 5 show the comparison of communication overhead between SESA and EPPKS. It can be seen that the SESA significantly reduces the communication overhead.

8.2. Extended SESA versus EPPKS

In this subsection, we will compare our extension of SESA with EPPKS [15] in terms of the computation overhead in the conjunctive keywords search process. Suppose there are 10 keywords tags on each bid and five keywords in the EB'_j conjunctive search trapdoor. In the extension, it costs the EB_j $10+1$ pairing operations and $10+2$ exponentiation operations to generate an energy encryption $\left(AC_j,S_j,C'_j\right)$, i.e., $11C_\mathrm{p}+13C_\mathrm{e}$. On the other hand, the DER_i or the RS needs $5+2$ hash operations and one exponentiation operation to compute the trapdoor. On receiving the trapdoor t'_w from RS, the local AS needs five pairing operations and one hash operation to test DER_i's query. If there is a suitable bid, the local AS needs to compute two pairings to verify the signature and one pairing to test if there is energy to satisfy DER_i's load demand. The DER_i or RS requires 1 pairing operation and 1 exponentiation operation to decrypt the ciphertext.

In comparison, the EPPKS needs $10+2$ pairing operations and $10*2+4$ exponentiation operations to generate an energy encryption on 10 keywords, i.e., $12C_\mathrm{p}+24C_\mathrm{e}$. Because EPPKS can do one keyword search at a time, for five-keyword search, the seeker needs to compute five trapdoors and sends them to the server, which needs five exponentiation operations. Thus, the server needs to test five times. Each time, the server needs one pairing operation to test whether a given tag contains keyword T_{w_i} or not. Thus, the server totally needs $5C_\mathrm{p}$ to test all of the trapdoors. If there is a matching item, the server needs $2C_\mathrm{p}+2C_\mathrm{e}$ more computation overhead to have an intermediate result of the partial decipherment. At last, it will cost the seeker C_e to recover the ciphertext.

In Table IV and Figure 6, it can be seen that the extension of SESA requires much less computation overhead than the EPPKS for the conjunctive keywords search. In addition, the extension is also more efficient than the EPPKS in terms of communication overhead because more trapdoors need to be sent to the server in EPPKS.

Table III. Comparison of communication complexity (bit).

Communication	SESA	EPPKS
EB to AS	803	640
DER to AS	160	160
AS to RS	160	1665

SESA, searchable encryption scheme for auction; EPPKS, efficient privacy preserving keyword search; EB, energy buyer; AS, auction server; DER, distributed energy resource; RS, registration server.

Table IV. Comparison of computation complexity.

Communication	SESA	EPPKS
EB	$11C_\mathrm{p}+12C_\mathrm{e}$	$12C_\mathrm{p}+24C_\mathrm{e}$
AS	$3C_\mathrm{p}$	$7C_\mathrm{p}+2C_\mathrm{e}$
DER_i or RS	$C_\mathrm{p}+2C_\mathrm{e}$	$6C_\mathrm{e}$

SESA, searchable encryption scheme for auction; EPPKS, efficient privacy preserving keyword search; EB, energy buyer; AS, auction server; DER, distributed energy resource; RS, registration server.

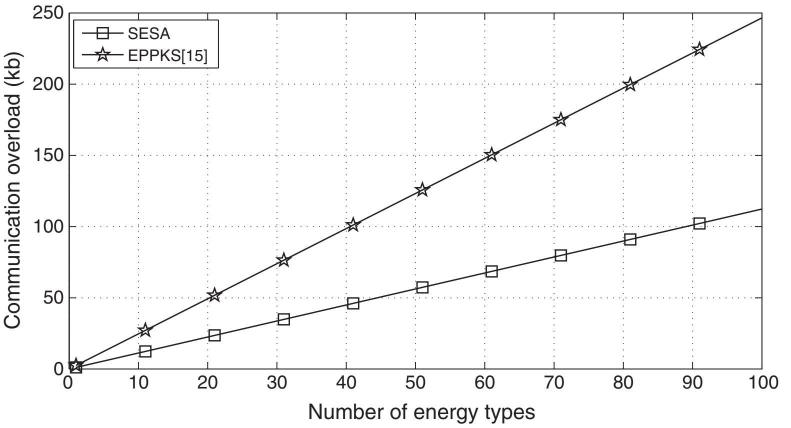

Figure 5. Comparison of communication overhead between searchable encryption scheme for auction (SESA) and efficient privacy preserving keyword search (EPPKS) schemes.

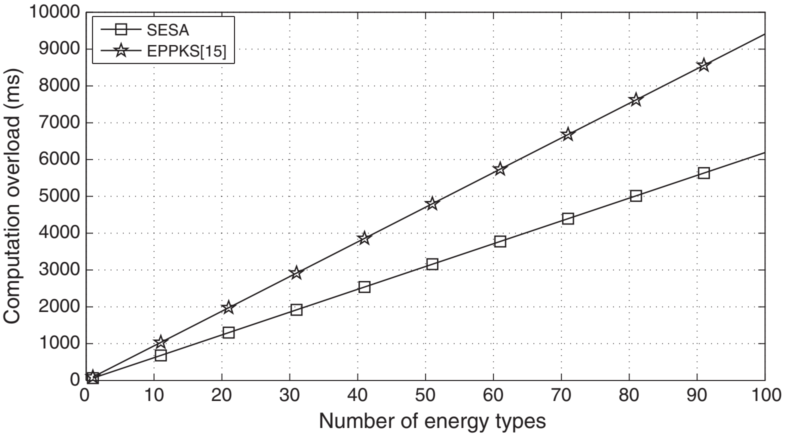

Figure 6. Comparison of computation between extended searchable encryption scheme for auction (SESA) and efficient privacy preserving keyword search (EPPKS) schemes.

9. CONCLUSION

In this paper, we have discussed the security and privacy concerns associated with energy auction in smart grid marketing and proposed an efficient SESA. We use PEKS to enable the energy sellers (DERs) to inquire potential winner from the AS while preserving the privacy of the EBs. In addition, an extension of SESA was presented to support detailed filtering of the bids. Security and performance analysis demonstrate that both our proposed SESA and its extension can achieve data and keyword privacy, bid integrity, and trapdoor unforgeability, and they are more efficient than the existing keyword search approach EPPKS in terms of computation and communication overhead. In the future, we will consider more complex conditions, such as efficient range search and filtering of energy trading in smart grid marketing.

ACKNOWLEDGEMENTS

This work is supported by the National Natural Science Foundation of China under grant no. 60903188, 61103207, and 61272437, and NSERC, Canada.

REFERENCES

1. Yuen C, Oudalov A, Timbus A. The provision of frequency control reserves from multiple micro-grids. *IEEE Transactions on Industrial Electronics* 2011; **587**(1): 173–183.
2. Ramachandran B, Srivastava SK, Edrington CS, *et al*. An intelligent auction scheme for smart grid market using a hybrid immune algorithm. *IEEE Transactions on Industrial Electronics* 2011; **58**(10): 4603–4612.
3. Forte VJ. Smart grid at national grid. In *Proceeding of ISGT 2010*, Gaithersburg, MD, Jan. 2010; 1–4.
4. Chakraborty S, Weiss MD, Simoes MG. Distributed intelligent energy management system for a single-phase high-frequency AC microgrid. *IEEE Transactions on Industrial Electronics* 2007; **54**(1): 97–109.
5. Lu R, Liang X, Li X, Lin X, Shen X. EPPA: an efficient and privacy-preserving aggregation scheme for secure smart grid communications. *IEEE Transactions on Parallel and Distributed Systems* 2012; **23**(9): 1621–1632.
6. Li H, Liang X, Lu R, Lin X, Shen X. EDR: an efficient demand response scheme for achieving forward secrecy in smart grid. In *Proceedings of 2012 IEEE Global Telecommunications Conference* (GLOBECOM 2012), to appear.
7. Chang YF, Chang CC. Enhanced anonymous auction protocols with freewheeling bids. In *Proceedings of 20th International Conference on Advanced Information Networking and Application* (AINA06), 2006; 353–358.
8. Li MJ, Justie STJ, Jennifer HCT. Practical electronic auction scheme with strong anonymity and bidding privacy. *Information Sciences* 2011; **181**(12): 2576–2586.
9. Shao J, Cao Z, Liang X, Lin H. Proxy re-encryption with keyword search. *Information Sciences* 2010; **180**(13): 2576–2587.
10. Bompard E, Lu W, Napoli R. Network constraint impacts on the competitive electrically markets under supply-side strategic bidding. *IEEE Transactions on Power Systems* 2006; **21**(1): 160–170.
11. Li X, Liang X, Lu R, Lin X, Zhu H, Shen X. Securing smart grid: cyber attacks, countermeasures and challenges. *IEEE Communications Magazine* 2012; **50**(8): 38–45.
12. Song Y, Ni Y, Wen F. An improvement of generation firm's bidding strategies based on conjectural variation regulation via dynamic learning. *In proceedings of the CSEE* 2003; **23**(12): 23–27.

13. Kanga DJ, Kimb BH, Hur D. Supplier bidding strategy based on non-cooperative game theory concepts in single auction power pools. *Electric Power Systems Research* 2007; **77**(5–6): 630–636.
14. Liaw HT, Juang WS, Lin CK. An electronic online bidding auction protocol with both security and efficiency. *Applied Mathematics and Computation* 2006; **174**(2): 1487–1497.
15. Liu Q, Wang G, Wu J. An efficient privacy preserving keyword search scheme in cloud computing. In *Proceeding of 2009 International Conference on Computational Science and Engineering*, 2009; 715–720.
16. Boneh D, Crescenzo DG, Ostrovsky R, Persiano G. Public key encryption with keyword search. In *Proceeding of EUROCRYPT* 2004. LNCS, vol. 3027, 506–522.
17. Fadlullah ZM, Kato N, Lu R, Shen X, Nozaki Y. Towards secure targeted broadcast in smart grid. *IEEE Communications Magazine* 2012; **50**(5): 150–156.
18. Fouda M, Fadlullah ZM, Kato N, Lu R, Shen X. A light-weight message authentication scheme for smart grid communications. *IEEE Transactions on Smart Grid* 2011; **2**(4): 675–685.
19. Lin X, Lu R, Foxton K, Shen X. An efficient searchable encryption scheme and its application in network forensics. In *Proceeding of E-Forensics* 2010, LNICST 56, 66–78.
20. Zhang B, Zhang FG. An efficient public key encryption with conjunctive-subset keywords search. *Journal of Network and Computer Applications* 2011; **34**(1): 262–267.
21. Dan B, Ben L, Hovav S. Short signatures from the Weil pairing. *Journal of Cryptology* 2004; **17**(4): 297–319.
22. Scott M. Efficient implementation of cryptographic pairings. Available: http://ecrypt-ss07.rhul.ac.uk/Slides/Thursday/mscottsamos07.pdf
23. Shamus software. http://www.shamus.ie/index.php?page=Benchmarks.
24. Libert B, Quisquater JJ. The exact security of an identity based signature and its applications. Preprint available at http://eprint.iacr.org/2004/102.
25. Galbraith SD. Pairings. *Advances in Elliptic Curve Cryptography*, Chapter 9, Cambridge University Press: New York, NY, USA; 2005; 183–213.