EDR: An Efficient Demand Response Scheme for Achieving Forward Secrecy in Smart Grid

Hongwei Li+, Xiaohui Liang+, Rongxing Lu+, Xiaodong Lin§, and Xueming (Sherman) Shen+
1School of Computer Science and Engineering, University of Electronic Science and Technology of China, China
1Department of Electrical and Computer Engineering, University of Waterloo, Canada
5Faculty of Business and Information Technology, University of Ontario Institute of Technology, Canada

Abstract—Compared with traditional power grid, smart grid has several distinguished features, i.e., distributed energy, large-capacity, robust to load fluctuations, and close consumer-grid interactions. Demand response is vital for smart grid, which is expected to save energy, maintain supply-demand balance, and reduce consumers’ electricity bills. Meanwhile, it is paramount to preserve consumers privacy and cyber security in smart grid. To tackle these challenging issues, in this paper, we propose an efficient demand response (EDR) scheme which utilizes the homomorphic encryption to achieve privacy-preserving demand aggregation and efficient response. Unlike existing schemes, the proposed EDR scheme can also achieve forward secrecy in addition to security features including confidentiality, authenticity and integrity. Extensive analysis demonstrates its security, and efficiency in terms of the computation and communication overhead.

Index Terms—Smart grid, Demand response, Forward secrecy.

I. INTRODUCTION

Recently, smart grid has emerged and recognized as the next generation of power grid [1], [2], due to its attractive characteristics, e.g., self-monitoring, self-healing, self-activating, and demand response (DR) [3], [4]. DR is a mechanism which assists consumers to use energy efficiently and transfer non-emergent power demand from on-peak to off-peak. DR can also bring various benefits to consumers. For example, consumers can reduce their electricity expenditure by matching the operation time of different electric appliances to the period with the best price. To enable the above features of DR, it often relies on a control center to implement real-time management of users’ electricity demand.

Though smart grid has many promising applications, security and privacy are very critical to success of smart grid [5], [6]. Adversaries might forge power consumption data at a large scale to attack smart grid, e.g. overloading power plants. Also, attackers might eavesdrop the communication, or even hack into power company’s database, to access the users’ consumption data from which they learn about the users’ habits, lifestyles daily activities and other privacy. To preserve user privacy and cyber security, the DR scheme should not only provide the electricity demand privacy, but also prevent the demand and response messages from being unauthorized access and modification.

Among many security and privacy requirements to protect demand and response messages in smart grid, forward secrecy is extremely important since cryptographic computations, e.g. encryption, signature and authentication, are often carried out on insecure devices. Forward secrecy [7] is to ensure that a short-lived session key derived from a set of long-term public and private keys will not be compromised even if one of the (long-term) private keys is compromised in the future. To mitigate the damage caused by exposure of secret keys stored on such devices, forward secrecy should be considered.

Despite its importance, forward secrecy has not been well studied in smart grid due to the complexity of smart grid communication. Existing schemes mainly focus on achieving confidentiality, data integrity and authenticity [8], [9], [10]. The first attempt to achieve forward secrecy in smart grid was tackled in [11]. However, since it adopts RSA public key algorithm and Diffie-Hellman exchange protocol to evolve the session key, the computation and communication overhead are heavy, thereby making it impractical.

In this paper, we propose an Efficient Demand Response scheme with forward secrecy, named EDR, to achieve privacy-preserving aggregation operations on the consumers’ electricity demand reports. Specifically, the contributions of this paper are twofold.

- Firstly, we propose the novel EDR scheme that utilizes homomorphic encryption [12] to achieve privacy-preserving demand aggregation and efficient response. The security analysis demonstrates that EDR can achieve forward secrecy, confidentiality, authenticity and data integrity.
- Secondly, we compare EDR with another scheme [11] which has the same security level. The comparison results demonstrate that EDR is more efficient in terms of the computation and communication overhead.

The remainder of the paper is organized as follows. In Section II, we formalize network model, security model and design goal. In Section III, we review the homomorphic encryption and bilinear pairing. Then, we propose the EDR scheme in Section IV, followed by the security analysis and performance evaluation in Section V and Section VI, respectively. We also present related works in Section VII. Finally, we draw our conclusions in Section VIII.
II. MODELS AND DESIGN GOAL

A. Network Model

As shown in Fig. 1, network model for smart grid is divided into a number of hierarchical networks comprising control center (CC), building area network (BAN) and home area network (HAN). CC covers n BANs. For the sake of simplicity, we assume each BAN comprises m HANs. Each HAN is assigned a smart meter enabling an automated, two-way communication between the CC and HAN users. Meantime, each BAN is equipped with a gateway (GW).

Communication between HAN user and the BAN GW (BG) is through relatively inexpensive WiFi technology, i.e., within the WiFi coverage of the BG, each HAN user can directly communicate with the BG. For the HAN user who is beyond the coverage of the BG, communication has to be done in multiple hops. However, since the distance between BG and CC is far away, the communication between BG and CC is through either wired links or any other links with high bandwidth and low delay.

![Network model for smart grid](image)

B. Security Model

In our security model, CC and BGs are trusted by all parties in the scheme, and infeasible for any adversary to compromise. In specific, we consider the following security goals needed to be achieved.

- **Forward secrecy:** It should be ensured that a session key derived from a set of long-term public and private keys will not be compromised if one of the (long-term) private keys is compromised in the future. If an adversary \( A \) compromises the HAN users or BGs, \( A \) cannot get any information for any previous electricity information. As a result, the forward secrecy can be achieved.

- **Privacy-preserving:** The users’ electricity demand should not be disclosed to the unauthorized/untrusted entities. Even if an adversary \( A \) hacks into the database of BGs and CC, it can also not identify the contents of ciphertext.

- **Confidentiality:** The electricity demand and response messages should be confidential, i.e., if the adversary \( A \) eavesdrops the communication all over the smart grid, it cannot identify the encrypted data.

- **Authenticity and data integrity:** BGs should authenticate HAN users for preventing illegitimated HAN users from joining the smart grid. Meanwhile, if the adversary \( A \) forges and/or modifies the electricity demand and response messages, the malicious operations can be detected.

C. Design Goal

Under the above models, our design goal is to develop a secure privacy-preserving demand response scheme for smart grid achieving forward secrecy. Specifically, the following two desirable objectives will be achieved.

- The proposed scheme should achieve efficient forward secrecy, i.e., the evolution of session key between HAN user and BG should be cost-effective in terms of computation and communication overhead.

- The proposed scheme should achieve the electricity demand privacy preservation, the demand's source authentication and data integrity, the confidentiality of the response messages.

III. PRELIMINARIES

In this section, we review the homomorphic encryption and bilinear pairing, which will serve as the basis of the proposed EDR scheme.

A. Homomorphic Encryption

Homomorphic Encryption (HE) allows certain algebraic operations on the plaintext to be performed directly on the ciphertext. HE is usually used for privacy-preserving operations (e.g. data aggregation, e-voting). In this paper, we adopt the Paillier cryptosystem [12]. In the Paillier cryptosystem, the public key is \( pk(n, g) \), and the corresponding private key is \( sk(\lambda, \mu) \). Let \( E(\cdot) \), \( m \), and \( r \) be the encryption function, a message and a random number, respectively. The ciphertext \( c \) is \( c = E(m) = g^m \cdot r^n \mod n^2 \). Then, the additive homomorphic property is as follows:

\[
E(m_1) \cdot E(m_2) = (g^{m_1} \cdot r^n)(g^{m_2} \cdot r^n) \mod n^2
= g^{m_1+m_2} \cdot (r^1r^n)^n \mod n^2
= E(m_1 + m_2)
\]

B. Bilinear Pairing and Identity-based signature

Let \( G \) and \( G_T \) be two multiplicative cyclic groups of the same prime order \( q \), and \( P \) be a generator of group \( G \). Suppose \( G \) and \( G_T \) are equipped with a pairing, i.e., a non-degenerated and efficiently computable bilinear map \( e : G \times G \rightarrow G_T \) such that \( e(aP_1, bQ_1) = e(P_1, Q_1) \) for all \( a, b \in \mathbb{Z}_q^* \) and any \( P_1, Q_1 \in G \). We refer to [13] for a more comprehensive description of pairing technique, and complexity assumptions.
A. CC Initialization

CC publishes the system parameters as $\text{params} = \{q, p, \mathbb{G}, \mathbb{G}_T, e, H_1, H_2\}$

- Keygen: Given an user’s identity $ID$, PKG computes $Q_{ID} = H_1(ID)$ and the associated private key $d_{ID} = sQ_{ID}$ that is transmitted to the user.
- Sign: In order to sign message $M$, the user picks a random number $r \in \mathbb{Z}_q^*$, and computes $U = rP + H_2(ID, M, U)$. The signature on $M$ is the pair $(\sigma = \langle U, V \rangle)$
- Verify: To verify a signature $(\sigma = \langle U, V \rangle)$ on a message $M$ for an identity $ID$, the verifier accepts the signature if $e(P, V) = e(P_{pub}, H_1(ID)) e(U, H_2(ID, M, U))$ and rejects it otherwise.

IV. PROPOSED EDR SCHEME

A. CC Initialization

For a single-authority smart grid under consideration, we assume a control center (CC) will bootstrap the whole system. Specifically, given the security parameter $\kappa$, CC first generates the bilinear parameters $(q, p, \mathbb{G}, \mathbb{G}_T, e)$ by running $\text{Gen}(\kappa)$, and chooses one secure symmetric encryption algorithm $\text{Enc}()$, i.e., AES, and three secure cryptographic hash functions $H_1, H_2, \mathbb{G}_T: \{0, 1\}^* \rightarrow \mathbb{G}$ and $H_3: \mathbb{G}_T \rightarrow \mathbb{G}_T^*$. In addition, CC also chooses a random numbers $\alpha \in \mathbb{Z}_q^*$, and computes $Q = \alpha P$ and $P_{CC} = \alpha H_1(ID_{CC})$, where $\mathbb{ID_{CC}}$ is the identity string of CC. CC also calculates the homomorphic encryption’s public key $(n, g)$, and the corresponding private key $(\lambda, \mu)$. Finally, CC publishes the system parameters as

$$\text{pubs} = \{q, p, \mathbb{G}, \mathbb{G}_T, e, Q, H_1, H_2, H_3, n, g, \text{Enc}()\}$$

B. BAN Initialization

Each HAN user $U_{ij} \in B_i(i = 1, 2, \ldots, n)$ chooses a random number $S_{BGi} \in \mathbb{Z}_q^*$ as its master key and computes its private point $P_{BGi} = S_{BGi} H_1(ID_{BGi})$. When a HAN user $U_{ij}$ registers itself in the $BG_i$, $BG_i$ runs the following steps:

1. $BG_i$ grants $SK_{U_{ij}}$ to $U_{ij}$.
2. After receiving $SK_{U_{ij}}$, $U_{ij}$ can non-interactively share a session key $K_{U_{ij}}$ with $BG_i$, $U_{ij}$ computes $K_{U_{ij}} = e(S_{BGi}, H_1(ID_{BGi}))$. $CC$ computes $K_{BGi} = H_3(e(P_{CC}, H_1(ID_{BGi})))$. The correctness is shown as follows:

$$K_{BGi} = H_3(e(S_{BGi}, H_1(ID_{BGi}))) = H_3(e(H_1(ID_{BGi}), H_1(ID_{CC}))) = H_3(e(P_{CC}, H_1(ID_{BGi})))$$

C. Demand Aggregation

Each HAN user $U_{ij} \in B_i(i = 1, 2, \ldots, n)$, $j = 1, 2, \ldots, m$ uses the smart meter to collect electricity demand $d_{ij}$, and performs the following steps:

1. $U_{ij}$ uses the homomorphic encryption to output the ciphertext $C_{U_{ij}}$ as mentioned in the III-A.
2. $U_{ij}$ uses the private key $SK_{U_{ij}}$ to make an identity-based signature $\sigma_{U_{ij}}$ on $M$ as mentioned in the III-B.
3. After receiving encrypted electricity demand $C_{U_{ij}}$, $BG_i$ checks signature $\sigma_{U_{ij}}$ to verify its validity as mentioned in the III-B. After the validity checking, $BG_i$ performs the following steps:
   a. $BG_i$ computes the aggregated demand $C_{BGi} = C_{BGi} \cdot C_{B1} \cdots C_{BGn}$.
   b. $BG_i$ uses the private key $SK_{BGi}$ to make an identity-based signature $\sigma_{BGi}$ on $M$ as mentioned in the III-B.
   c. $BG_i$ sends the aggregated results $C_{BGi} = C_{BGi} \cdot [\sigma_{BGi} \cdot \sigma_{U_{ij}}]$ to the CC.

D. Demand Processing and Response

Upon receiving $n$ encrypted electricity demand $C_{BGi} = C_{BGi} \cdot [\sigma_{BGi}(i = 1, 2, \ldots, n)]$, $CC$ first checks signature $\sigma_{BGi}$ to verify its validity as mentioned in the III-B. Then CC performs the following steps to read the aggregated demand $C$, where $C$ is implicitly formed by $C = \prod_{i=1}^{n} C_{BGi} = \prod_{j=1}^{m} C_{U_{ij}}$.

1. $CC$ uses the homomorphic encryption’s private key $(\lambda, \mu)$ to decrypt $C$ and $C_{BGi}(i = 1, 2, \ldots, n)$ as mentioned in the III-A. The total electricity demand and $BG_i$’s aggregated electricity demand are $\sum_{i=1}^{n} (\sum_{j=1}^{m} d_{ij})$ and $\sum_{j=1}^{m} d_{ij}$, respectively.
• Step-2: After analyzing the real-time electricity demand \( \sum_{i=1}^{n} d_{ij} \) and \( \sum_{j=1}^{m} d_{ij} \) for \( i = 1, 2, \ldots, n \), CC generates the response message \( S_i \) for \( BG_i \) (\( i = 1, 2, \ldots, n \)), respectively [3, 4].

• Step-3: CC sends \( C_i = \text{Enc}_{K_{\text{cc}}(S_i || CC || BG_i || TS)} \) to \( BG_i \) (\( i = 1, 2, \ldots, n \)), respectively, where \( C_i = \text{Enc}_{K_{\text{cc}}(S_i || CC || BG_i || TS)} \) and \( TS \) is the current timestamp.

• Step-4: Upon receiving \( C_i || CC || BG_i || TS \), \( BG_i \) decrypts \( C_i \) to get \( S_i \). Then \( BG_i \) forwards \( C_{ij} = \text{Enc}_{K_{ij} \cdot BG_i} (S_{ij} || BG_i || U_{ij} || TS), TS \) is the current timestamp.

• Step-5: After receiving \( C_i || CC || BG_i || TS \), HAN user \( U_{ij} \) decrypts \( C_{ij} \) to get \( S_{ij} \). Then \( U_{ij} \) analyzes \( S_{ij} \) and determines to shift power use from peak times to non-peak times for lower electricity bills [3, 4].

E. Session Keys Evolving

HAN user \( U_{ij} \) (\( i = 1, 2, \ldots, n, j = 1, 2, \ldots, m \)) can evolve its private key \( SK_{U_{ij}} \) by performing the following steps.

• Step-1: \( U_{ij} \) chooses a random number \( r_{ij} \in \mathbb{Z}_p^* \) and computes \( Q_{U_{ij}} = r_{ij} H_1(ID_{BG_i}), \) and its short-lived private key \( SK_{U_{ij}} = r_{ij} S_{KU_{ij}}, \) where \( SK_{U_{ij}} \) is its long term identity-based private key.

• Step-2: \( U_{ij} \) sends \( C_{ij} = \text{Enc}_{K_{ij} \cdot BG_i} (Q_{U_{ij}} || BG_i || U_{ij} || TS), TS \) is the current timestamp.

• Step-3: After receiving \( C_{ij} || BG_i || U_{ij} || TS \), \( BG_i \) decrypts \( C_{ij} \) to get \( Q_{U_{ij}} \) with \( K_{U_{ij} \cdot BG_i} \) and computes the new session key

\[
K_{U_{ij} \cdot BG_i} = H_3(e(H_1(ID_{U_{ij}}), S_{BG_i} \cdot Q_{U_{ij}})) \tag{5}
\]

Then \( BG_i \) sends \( \text{Enc}_{K_{U_{ij} \cdot BG_i}} (Q_{U_{ij}} || BG_i || U_{ij} || TS) \) to the HAN user \( U_{ij} \), where \( TS \) is the current timestamp. Finally, \( BG_i \) deletes the previous session key and session messages.

• Step-4: After receiving \( \text{Enc}_{K_{U_{ij} \cdot BG_i}} (Q_{U_{ij}} || BG_i || U_{ij} || TS) \), \( U_{ij} \) firsts receives \( Q_{U_{ij}} || BG_i || U_{ij} || TS \) with the new session key

\[
K_{U_{ij} \cdot BG_i} = H_3(e(SK_{U_{ij}}, H_1(ID_{BG_i}))) \tag{6}
\]

Then \( U_{ij} \) checks if the recovered \( Q_{U_{ij}} \) is the same as the local \( Q_{U_{ij}} \). If the check is positive, \( U_{ij} \) deletes \( Q_{U_{ij}}, SK_{U_{ij}}, \) the previous session key and session messages. The correctness is shown as follows:

\[
K_{U_{ij} \cdot BG_i} = H_3(e(SK_{U_{ij}}, H_1(ID_{BG_i})))
= H_3(e(H_1(ID_{U_{ij}}), H_1(ID_{BG_i}))) r_{ij} S_{KU_{ij}}, \tag{7}
= H_3(e(H_1(ID_{U_{ij}}), S_{BG_i} \cdot Q_{U_{ij}}))
\]

V. SECURITY ANALYSIS

In this section, we analyze the security properties of the proposed EDR scheme. Especially, following the security model discussed earlier, we are most concerned with how the proposed EDR scheme can achieve the forward secrecy of session between HAN user and BG, the individual HAN user’s electricity demand privacy preservation, the demand’s source authentication and data integrity, and the confidentiality of the response messages.

A. The proposed EDR scheme provides the forward secrecy of session between HAN user and BG

In the proposed EDR scheme, the confidentiality of session between HAN user and BG is achieved by the secure session key. Furthermore, in the session key evolving phase, \( BG_i \) will delete the previous session key and session messages after it computes the new session key \( K_{U_{ij} \cdot BG_i} = H_3(e(H_1(ID_{U_{ij}}), S_{BG_i} \cdot Q_{U_{ij}})) \). Therefore, \( U_{ij} \) will also delete the previous session key and session messages after it recovered \( Q_{U_{ij}} \) with the new session key \( K_{U_{ij} \cdot BG_i} = H_3(e(SK_{U_{ij}}, H_1(ID_{BG_i}))) \). Thus, exposure of a secret key corresponding to a given time period does not allow the adversary \( A \) to get session messages for any prior time period. As a result, even if \( A \) compromises the HAN users or BGs, \( A \) cannot get any previous session key. Therefore, the forward secrecy of session between HAN user and BG is achieved in the proposed EDR scheme.

B. The proposed EDR scheme provides the electricity demand privacy

In the proposed EDR scheme, since HAN user’s electricity demand is a homomorphic encryption ciphertext [12], the adversary \( A \) cannot identify the corresponding electricity demand even though \( A \) eavesdrops the ciphertext. Moreover, since \( BG_i \) only aggregates and does not decrypt the electricity demands, \( A \) cannot get the electricity demand even if \( A \) compromises the \( BG_i \’s \) database. Finally, \( C-C \) recovers the aggregated demands \( \sum_{i=1}^{n} \sum_{j=1}^{m} d_{ij} \) and \( \sum_{j=1}^{m} d_{ij} \) for \( i = 1, 2, \ldots, n \). However, since \( \sum_{i=1}^{n} \sum_{j=1}^{m} d_{ij} \) and \( \sum_{j=1}^{m} d_{ij} \) are all aggregated results, even if \( A \) intrudes the CC’s database, \( A \) still cannot get the each HAN user’s electricity demand. Therefore, the proposed EDR scheme provides the electricity demand privacy.

C. The proposed EDR scheme provides the demand’s source authentication and data integrity

In the proposed EDR scheme, each HAN user’s electricity demand and the aggregated demand are signed by identity-based signature [14]. Since the signature is provably secure in the random oracle model [14], the source authentication and data integrity can be guaranteed.

D. The proposed EDR scheme provides the confidentiality of response messages

In the proposed EDR scheme, when CC sends the response messages to \( BG_i \) (\( i = 1, 2, \ldots, n \)), CC encrypts them as \( C_{ij} = \text{Enc}_{K_{U_{ij} \cdot BG_i} (S_{ij} || BG_i || U_{ij} || TS)} \). Then, \( BG_i \) forwards the response messages to \( U_{ij} \) (\( j = 1, 2, \ldots, m \)) in the form of \( C_{ij} = \text{Enc}_{K_{U_{ij} \cdot BG_i} (S_{ij} || BG_i || U_{ij} || TS)} \), where \( C_{ij} \) are encrypted by AES. Since AES is semantic secure, the response
messages achieve the confidentiality in the proposed EDR scheme.

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Confidentiality</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Authenticity</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Data integrity</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Forward secrecy</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
</tbody>
</table>

We present the comparison results of security level in Table I. It can be seen that the scheme [8] only achieves the confidentiality, the scheme [9] achieves confidentiality and data integrity, the scheme [10] achieves confidentiality, data integrity and authenticity, and the scheme [11] and the proposed EDR scheme both achieve confidentiality, data integrity, authenticity and forward secrecy.

VI. PERFORMANCE EVALUATION

In this section, we evaluate the computation and communication overhead of the session key evolving between user $U_{ij}$ and $BG_i$ in both EDR and the scheme [11].

A. Computation Overhead

Compared to exponentiation operations in $G$, pairing operations and RSA encryption/decryption, the computation overhead of AES encryption/decryption and hash operations are negligible [15]. In EDR, as described in IV-E, it requires 2 multiplication operations in $G$ for generating $Q_{U_{ij}}$ and the next time period’s private key $SK_{U_{ij}}$. After receiving the message from $U_{ij}, BG_i$ computes the new session key with 1 pairing operation and 1 multiplication operation in $G$. Then $U_{ij}$ computes the new session key with 1 pairing operation. Denote the computation overhead of a multiplication operation in $G$ and a pairing operation by $C_m$ and $C_p$, respectively. Thus, the total computation overhead is $3 + C_m + 2 * C_p$.

In comparison, for the scheme [11], it requires 1 RSA encryption for HAN user $i$ to generate the request packet. After receiving the ciphertext from HAN user $i$, $BG_j$ decrypts the request packet including 1 RSA decryption and computes the new session key with 1 exponentiation operation in $Z_q^*$. In addition, $BG_j$ sends an encrypted response message including 1 RSA encryption. Then HAN user $i$ decrypts the response message with 1 RSA decryption and computes the new session key including 1 exponentiation operation in $Z_q^*$. Denote the computation times of an exponentiation operation in $Z_q^*$, a RSA encryption and a RSA decryption by $C_e, RSA_e$ and $RSA_d$, respectively. Therefore, the total computation overhead is $2 * (C_e + RSA_e + RSA_d)$. Experiments were conducted on a Pentium IV 3GHz system to study the execution time [16], [17]. For $G$ over the FST curve, a single multiplication operation costs 1.0 ms and the corresponding pairing operation costs 2.8 ms. Meantime, a 1024-RSA decryption and a 1024-RSA encryption cost 3.88ms and 0.02ms, respectively. An exponentiation operation in $Z_q^* (|q| = 1024)$ costs 0.64ms.

The comparison of computation overhead is shown in Fig.2. We can see that EDR achieves lower execution times compared to the scheme [11].

B. Communication Overhead

In EDR, $U_{ij}$ firstly sends message to $BG_i$. The message is in the form of $C_{U_{ij}} | BG_i | U_{ij} | TS$, where $C_{ij} = Enc_{K_{ij}} ([Q_{U_{ij}} || BG_i || U_{ij} || TS])$. If we choose AES ciphertext with 256-bit, $G$ with 160-bit order, and set $|BG_i| + |U_{ij}| + |TS|$ as 80-bit length, The whole message size will be $256 + 80 = 336$ bits. Then, $BG_i$ responds a message in the form of $Enc_{K_{U_{ij}}} - BG_i (BG_i || U_{ij} || TS)$ and its size is 256 bits. So the total communication overhead is $336 + 256 = 592$ bits.

In comparison, in the scheme [11], HAN user $i$ first sends a RSA ciphertext to $BG_j$ in the form of $\{i||j||g^n\}_\{encr\}Pub_{HAN,CW_j}$. Then BAN $GW_j$ decrypts a RSA ciphertext to HAN user $i$ in the form of $\{i||j||g^n\}_\{encr\}Pub_{HAN,CW_j}$. Finally, HAN user $i$ sends a AES ciphertext to $BG_j$ in the form of $\{M_i||T_i||MAC_{K_i} ||(encr)_{K_i}\}$. The overall communication overhead consists of two RSA ciphertexts and one AES ciphertext. Therefore, the overall communication overhead will increase to $2 * 1024 + 256 = 2304$ bits if we choose 1024-bit RSA and 256-bit AES.

Fig. 3 plots the communication overhead at a given BG for different number of HAN users. When the number of HAN users is small, there are low communication overhead in both EDR and the scheme [11]. Then the communication overhead increases with the increased HAN users. However, It should be noted that the increase is much faster in the case of the scheme [11]. For instance, when 300 HAN users are considered for the given BG, the communication overhead for EDR is 173Kb, but 675Kb for the scheme [11]. EDR largely reduces the communication overhead for the session key evolving.

VII. RELATED WORKS

Homomorphic encryption can achieve certain algebraic operations on the plaintext to be performed directly on the ciphertext and has been used in many data aggregation schemes [18], [19]. Theses schemes are very promising and have
triggered considerable following research work [8], [9], [10], as reviewed below.

Li et al. [8] presented a distributed incremental data aggregation approach. To protect user privacy, homomorphic encryption is used to secure the data en route. Seo et al. [9] proposed a secure and efficient power management mechanism leveraging a homomorphic data aggregation and capability-based power distribution. The proposed mechanism enables to gather the power demands of consumers securely. Lu et al. [10] proposed a privacy-preserving aggregation scheme for secure and efficient smart grid communications. It realizes multi-dimensional data aggregation approach based on the homomorphic Paillier cryptosystem. Those schemes assumed that the session keys between HAN users and BG are unchanged. However, once an adversary $A$ compromises the session keys, $A$ can decrypt any previous response message. As a result, how to design the forward secure scheme for smart grid is a challenging issue. Fouda et al. [11] proposed a lightweight message authentication scheme achieving forward secrecy. Specifically, in the proposed scheme, the HAN users can achieve mutual authentication with BG. Detailed security analysis shows that the proposed scheme can satisfy confidentiality, data integrity, authenticity and forward secrecy. Since the scheme adopts RSA public key algorithms and Diffie-Hellman exchange protocol to evolve the session key between HAN users and BG, the computation and communication overhead are heavy. In this paper, we have designed an efficient forward secure scheme, where the session keys are evolved through AES symmetric encryption and decryption. Since AES is more lightweight than RSA, our approach can achieve lower computation and communication overhead compared with [11].

VIII. CONCLUSION

In this paper, we have proposed an efficient demand response (EDR) scheme with forward secrecy. It realizes HAN users’ electricity demand aggregation approach based on the homomorphic encryption and efficient response. Security analysis has demonstrated that the proposed EDR scheme can achieve forward secrecy, confidentiality, authenticity and data integrity. Performance evaluation further demonstrates its efficiency in terms of computation and communication overhead. In the future work, we will explore other challenging security issues in smart grid, such as the bogus messages attacks and denial of service attacks.

ACKNOWLEDGMENT

This work is supported by the National Natural Science Foundation of China under Grants 61103207, 61073106, and 61003232, the Fundamental Research Funds for Chinese Central Universities under Grant ZYGX2011J059, and NSERC, Canada.

REFERENCES


