This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TVT.2016.2577018, IEEE Transactions on Vehicular Technology



# Lightweight Authentication and Privacy-Preserving Scheme for V2G Connections

Asmaa Abdallah, and Xuemin (Sherman) Shen, *Fellow, IEEE*

***Abstract*—Vehicle-to-grid (V2G) connection allows two-way electricity transmission between electric vehicles (EVs) and power grid for achieving many known benefits. However, V2G connections suffer from certain security threats, such as EV's privacy and authenticating it to the grid. In this paper, we propose a lightweight secure and privacy-preserving V2G connection scheme, in which the power grid assures the confidentiality and integrity of exchanged information during (dis)charging electricity sessions and overcomes EVs' authentication problem. The proposed scheme guarantees the financial profits of the grid and prevents EVs from acting maliciously. Meanwhile, EVs preserve their private information by generating their own pseudonym identities. In addition, the scheme keeps the accountability for the electricity-exchange trade. Furthermore, the proposed scheme provides these security requirements by lightweight overhead; as it diminishes the number of exchanged messages during (dis)charging sessions. Simulation results demonstrate that the proposed scheme significantly reduces the total communication and computation load for V2G connection, especially for EVs.**

***Index Terms*—BlueJay ultra-lightweight hybrid cryptosystem; V2G connections; EVs.**

## I. Introduction

Smart grid utilizes various communication techniques to connect different parties in the gird to enhance grid's efficiency and reliability, i.e., optimal utilization of generated power and reduction of electricity losses. Thus, many short-term energy storage devices, such as fuel cells, flywheels, and electric vehicles (EVs), are used to save the extra electricity in case of high power generation and provide the electricity back to the grid in high consumption time. The EVs' batteries are considered promising storage media, because the number of EVs in the market increases rapidly and is expected to increase more in the near future. EVs' batteries are stable storage units; their losses ratio for stored power is low. In addition, the (dis)charging operations for EVs' batteries are performed much faster than increasing/decreasing the generation level of traditional power plants to satisfy the change in electricity demand. EVs quickly supply electricity to the grid if consumers' demands increased. They also can rapidly store the extra power from the grid if the electrical requirements decreased. Consequently, the vehicle-to-grid (V2G) connection term is coined to represent the bidirectional communication between EVs and power grid [2]–[4].



A. Abdallah and X. Shen are with the Department of Electrical and Computer Engineering University of Waterloo, Ontario, N2L 3G1, Canada. e-mail: a3abdall, xsheng@bbcr.uwaterloo.ca.

A preliminary version of this paper was presented at the 2015 IEEE GLOBECOM Conference [1].

*The architecture of V2G connection* consists of clusters of EVs located in the place; they are connected to devices called aggregators. The function of aggregators is to manage the power exchange operation between the grid and EVs. There are two main types of aggregators: local aggregators (LAs) and central aggregators (CAs). LAs are located in the same range of EVs' cluster to collect information about EVs, such as their batteries' state of charges (s.o.c), and forward the current electricity price from the grid's operator to them. LAs also receive EVs' requests to (dis)charge their batteries and compute the payment for each EV. On the other hand, CAs work as a link between the power grid's operators and EVs; they receive grid's requests to supply/consume electricity and the current price. They also control LAs to satisfy the requirements for both power grid and EVs' owners. Generally, EVs connect to LAs during their parking time, which is approximately $95\%$ of the day, so that LAs and charging stations (CSs) are mainly installed in the parking lots, such as residential buildings or companies' parking lots. In addition, V2G connection should contain representatives for the power grid, i.e., the grid operators; their function is to communicate with EVs via CAs to satisfy the grid's requirements [2]–[7].

Although V2G connection can solve the electricity losses problem and offer a fast supply/store electricity service to power grid, there are certain security and privacy concerns. V2G networks are vulnerable to attacks that threaten EV owners' and location privacy, EV's authenticity, and exchanged messages' confidentiality and integrity. First concern is EVs' privacy; EV owners' personal information should be protected from different parties. Intercepting EV's sensitive information, such as identity, account details, battery capacity, and s.o.c value, helps attackers to identify users and easily link between the EV and specific locations, and consequently extract information about user's life style, social relations, or health conditions. In addition, malicious adversaries or even legitimate entities can misuse owners' information for their benefits, such as a malicious LA tries to sell user's profile to other parties or decrease the profit of that user. Second concern is the EV's authenticity; V2G connections could also suffer from impersonation attacks, where a malicious party masquerades an innocent user's identity to acquire financial gains or cause disruptions in the charging processes. For example, a dishonest EV or a malicious adversary could impersonate an innocent EV to get financial gains or reduce their expenses, i.e., recharging for free or receiving incentives for ancillary services offered by the victim. Moreover, attackers may impersonate LAs to control the charging process and increase their profits. Finally, integrity attacks can tamper the exchanged messages among participated parties. Modify, delay

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TVT.2016.2577018, IEEE Transactions on Vehicular Technology



or replay a legitimate message or insert illegitimate messages can threaten messages' integrity, which cause service delay or disruption and consequently impact on grid's stability and efficiency. For example, an adversary changes a message sent by an innocent EV from "charge" to "discharge", which may lead to an empty battery for that EV [3], [12]–[14], [22], [23].

To accomplish their functions and resist theses attacks, V2G connections should satisfy their main security conditions utilizing different techniques. They may employ anonymity and non-linkability mechanisms to conceal EV owners' real identities from being revealed to illegal parties, also protects EV's multiple (dis)charging sessions from being linked together or to its real identity. Also, location-privacy mechanisms should be applied to prevent adversaries from link the EV with a specific location. However, controlled link between EV's different (dis)charging sessions should be existed to ensure accountability and traceability in case of a dispute or security incident. In addition, V2G connections use various authentication mechanisms to guarantee that the participated EV is the one that it claims to be, i.e., authenticating EVs to grid's operator; so, only authorized EVs can access network's resources and (dis)charge their batteries. As a result, impersonating EV to gain financial benefits is prevented and a fair billing is provided to each EV. Also, message authenticity and integrity should be assured using, for instance, keyed hash values or message authentication code. In other words, before any (dis)charging or payment operation, the grid needs to authenticate EVs, while EVs require to preserve their private information from grid's operator or any intermediate devices in the connection, e.g., LAs [3], [8]–[10], [12]–[14], [22], [23].

In this paper, we propose a lightweight secure authentication and privacy-preserving V2G connection scheme. The scheme allows EVs to generate their own pseudonym identities and do not expose their private information to any party even the grid's operator; so, the EVs' privacy is preserved. In addition, the scheme forces EVs to follow a specific procedure, i.e., finishing their part in the connection first, to prevent them from acting maliciously so that the scheme keeps the grid's financial profits. The scheme also assures confidentiality and integrity of exchanged messages during (dis)charging sessions in addition to keep the accountability of electricity-trade operation. Furthermore, the scheme achieves these security requirements with lightweight computation and communication overhead. The contributions of our paper can be listed as follows:

- A privacy-preserving scheme for EVs' owner and location information that based on allowing EVs to generate their own pseudonym identities and do not expose their private information to any party in the connection.

- A protection mechanism that forces the EVs to follow a specific procedure to prevent them from misbehaving, i.e., overcome EVs' authentication problem, and consequently guarantees the grid's profit.

- A tracing method to keep the accountability of the electricity-trade operation.

- In addition to assure confidentiality and integrity of the exchanged messages among different parties in the connection and analyze performance's efficiency of the proposed scheme in terms of communication and computation complexities.

The remaining of the paper is organized as follows. Section II discusses related works and existing solutions. Section III introduces our system model, security requirements, and design goals. Section IV reviews BlueJay ultra-lightweight hybrid cryptosystem. In Section V, we present our proposed scheme. Section VI gives security analysis, while Section VII evaluates the performance of our scheme. Finally, Section VIII concludes the paper.

## II. Related Work

The significant security concerns for V2G connection are securing the service provider, i.e., EV's privacy, and authenticating it to the grid. Before any (dis)charging or payment operations, the grid should authenticate EV, while EV requires preserving its private information from the grid's operator or any intermediate devices in the connection. Some research works only utilize EVs as temporary storage for cluster of consumers [11], while others focus on the security threats of V2G connection. Several solutions are proposed for these threats based on various procedures, such as different authentication mechanisms, anonymization techniques, physical security methods, and encryption schemes. For instance, elliptic curve cryptography (ECC) public key scheme is utilized in [13] to secure the multi-domain architecture V2G network, where each domain contains one LA and some EVs. While, in [14], the authors utilize combination of several cryptography schemes to propose a new security and privacy (V2GPriv) scheme for V2G network. V2GPriv utilizes cryptographic techniques, such as double encryption to conceal the different parties' communication relations from eavesdroppers. In addition, every EV's energy meter has a unique identifier and a public-private key pair for signing purposes; these secret parameters are assigned to the EV during the manufacturing and stored in the meter in a secure module. The $k$-anonymity service is also utilized so that the assigned pseudonym identity for the gateway is shared with $k-1$ other gateways. Mutual authentication operation is performed by basic challenge-response protocol to prove that the EV's meter has not been tampered with.

- *Authentication Schemes:* Some studies utilize authentication schemes to guarantee the validity of connected EVs. A unique batch authentication protocol for vehicle-to-grid (UBAPV2G) communications is proposed in [15]; UBAPV2G employs batch authentication to guarantee privacy and integrity of exchanged messages between LA and EVs. LA broadcasts a request message to EVs in the range; EVs respond by messages with their identification information. LA collects EVs' responses during a specific time interval, and then verifies the authenticity of them by one verification process using batch authentication. Batch authentication is better than one-by-one authentication method, as it saves the communication and computation overhead [16]. However, UBAPV2G scheme suffers from some security flaws that could be exploited by adversaries or dishonest aggregators [17]. While in [18], EVs' battery statuses are divided into charging, fully charged, and discharging states; the proposed scheme performs mutual authentication between EV and LA at each stage to confirm

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TVT.2016.2577018, IEEE Transactions on Vehicular Technology



EVs' legitimacy. According to [19], EVs' security concerns are classified according to their roles in V2G connection, i.e., customers, storage units, or generators, and a role-dependent privacy-preserving scheme with anonymous authentication is proposed.

- *Anonymization Mechanisms:* Other studies attempt to preserve the privacy of EVs' owners by anonymizing the vehicle so that outsiders cannot extract any distinguishable private information, such as owner's identity or EV's location. In [20], CA utilizes partially blind signature to issue a one-day permit for each EV, which uses the permit and its pseudorandom identity to establish a communication session with its LA. EV has to send periodic status reports to LA in addition to its (dis)charging requests. When the grid requires supplying/consuming electricity, it sends a public request and CAs directly make bids. If its bid successes, CA sends to LAs to select number of EVs that satisfies the required demand. EV's privacy is guaranteed, but if any EV cheats, CA exposes its real identity. A privacy-preserving protocol based on certificate less public key scheme and partially blind signature is proposed in [21]. Each EV has a partially private key from the trusted party and adds a random secret value to obtain its private key. So, the scheme reduces the dependence on the third party. In [22], authors suggest a security framework, where each EV acts as an independent agent with trusted platform module, which provides the required cryptographic parameters to anonymize the EV. However, an adversary model is presented in [23] to proves that any attacker can predict (by some probability) the identity of EV from CSs' locations and the distance between them even when EV uses several pseudo identities.

- *Physical Layer Protection Methods:* Other category of proposed solutions for V2G connection is based on physical layer protection mechanisms. The seamless connectivity of vehicular network could threaten EVs' security. So, the smart grid's devices, e.g., smart meter and smart appliances, are utilized in [24] to secure the EV by keeping it connected to the house whatever its location. In [25], the authors propose a physical method to resist the distributed DoS and jamming attacks using a channel-based key management approach to setup a secret key between two remote nodes by exchanging several beacon signals.

- *Other Encryption Schemes:* Other research works study the impact of price information and propose secure payment management schemes. In [26], the authors analyze the optimal charging polices for EVs using Markov chain considering EV's mobility and real-time price. A payment scheme is proposed in [27], where EV's owner first opens an account with the grid's operator. Before any (dis)charging operations, the supplier verifies the balance of EV's account, and performs the operation only if it satisfies. The proposed scheme uses bilinear pairing and decisional Diffie-Hellman assumption to generate the keys' parameters and partially blind signature scheme and zero-knowledge proof to preserve user's privacy.

In summary, V2G connection's security concerns can be confined to EV's owner and location privacy, EVs' authentication, and information integrity and confidentiality. Several solutions are proposed to deal with these concerns based on authentication, anonymitization, physical layer protection, or encryption methods. Nevertheless, these provided techniques suffer from certain back-draws, such as high communication and computation burden, or usage of special hardware devices.

## III. System Model

### A. Network Model

Our V2G network consists of a control center (CC), which belongs to the utility company; it is a trusted party. CC is connected to several LAs. LAs do not belong to utility company; they may be owned by independent distribution companies. They are located at local substations or distribution lines in the neighbour region. Each LA connects to fleets of EVs, which are located in different parking lots in the area. LA is also a trusted party; LAs = $\{LA_1, LA_2, \ldots, LA_m\}$, where $m$ is the number of substations or distribution lines. CC communicates with LAs through wired connection, e.g., Internet. In each parking lot, there is a cluster of EVs; EVs = $\{EV_1, EV_2, \ldots, EV_n\}$, where $n$ is the number of EVs in the cluster, so $n$ could be different for each lot. EV is a non-trusted party. The EVs' cluster is connected to LA via an access point (AP), which works as a relay node to forward the exchanged messages between LA and EVs through WiFi connection. LA connects to several CSs wirelessly via APs too. CSs may locate in the parking lots or specific charging place; CSs = $\{CS_1, CS_2, \ldots, CS_j\}$, where $j$ is the number of CSs controlled by LA. Keying parameters are provided to different parties by independent trusted authority (TA). V2G system model is shown in Fig 1.

### B. Adversary Model and Security Requirements

CC and LAs are honest but curious. They will not try to act maliciously toward EVs, but they may attempt to extract EVs' private information from exchanged messages. According to EVs, they are non-trusted parties. Some EVs may act selfishly to gain a benefit or prevent other EVs from obtaining advantages; also, malicious EVs may try to impersonate innocent EVs. In addition, malicious adversaries threaten V2G connection; adversary $\mathscr{A}$ can eavesdrop the exchanged messages between LA and EVs. Moreover, $\mathscr{A}$ may establish some active attacks; such as attempting to fabricate the captured messages, or begin a replay attack. Moreover, $\mathscr{A}$ may try to impersonate an honest EV to seize its connection with LA. To thwart these malicious actions, our proposed scheme will fulfill the following security requirements:

- Authentication: secure LA's messages against any unauthorized action, i.e., prevent any illegal parties from accessing or modifying LA's messages.

- EV's Privacy: assure that the private information of EV is not revealed to other parties and guarantee that nobody can link between EV's location and its owner's identity. Neither attackers nor CC and LAs can gain any distinguishable knowledge about a particular EV. So, they cannot link EV's battery status or location with the owner's identity.

- Confidentiality and Messages Integrity: guarantee that EVs' (dis)charging and service payment messages are confidential; they are only accessible by legitimate parties, e.g., LA and related EV. The messages' integrity should also be

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TVT.2016.2577018, IEEE Transactions on Vehicular Technology



guaranteed. Even if $\mathscr{A}$ already intercepts the message, he/she has no access to the key to decrypt it. Additionally, $\mathscr{A}$ cannot resend a message or modify its contents.

- Accountability: The previous electricity trade sessions should be traceable; CC guarantees the accuracy of former processes. No malicious LAs or EVs can forge previous bills to increase their profits.

### C. Design Goals

The main objectives of our proposed scheme can be divided into two folds:

- It should guarantee the security requirements for V2G connection. EVs owners' and location privacy should be preserved, and information confidentiality and integrity should be assured. Likewise, authentication of different parties should be guaranteed. Finally, accountability of the electricity trade operation should also assured.

- It also should be efficient and lightweight due to communication and computation overhead.

## IV. Preliminaries

We utilize BlueJay ultra-lightweight hybrid cryptosystem [28], which is a lightweight fast cryptosystem especially in its encryption process. BlueJay combines PASSERINE public key cryptosystem [29] and Hummingbird-2 lightweight symmetric scheme [30].

### A. PASSERINE cryptosystem

PASSERINE scheme is a lightweight version of Rabin public-key scheme [31] that has two advantages: the message space is fully utilized, and the encryption operation provides lightweight computation load.

*1) Key generation:* Generate two large random distinct primes $p$ and $q$ with roughly the same size as private key parameters. Then, compute the public key $n = pq$.

*2) Encryption:* Generate a set of co-prime numbers $b_1, b_2, \ldots, b_k$ denoted as a base and their product is $B = \prod_{i=1}^{k} b_i$. Encrypt the plaintext $m$ as:

$$C_0 \equiv m^2 + Y\ n (mod\, b_0),$$
$$C_1 \equiv m^2 + Y\ n (mod\, b_1),$$
$$\ldots\ldots\ldots,$$
$$C_m \equiv m^2 + Y\ n (mod\, b_m).$$

*3) Decryption:* Reconstruct the message $C$ as:

$$C = \left( \sum_{i=1}^{k} C_i . \frac{B}{b_i} . \left( \frac{B}{b_i} \right)^{-1}_{b_i} \right) mod\, B.$$

Compute $m_p = (c^{((p+1)/4)} mod\ p).q.q_p^{-1}$, and $m_q = (c^{((q+1)/4)} mod\ q).p.p_q^{-1}$. Select the right root from $m = \{m_p + m_q, m_p - m_q, -m_p + m_q, -m_p - m_q\}\ (mod\ n)$ as the plaintext message.

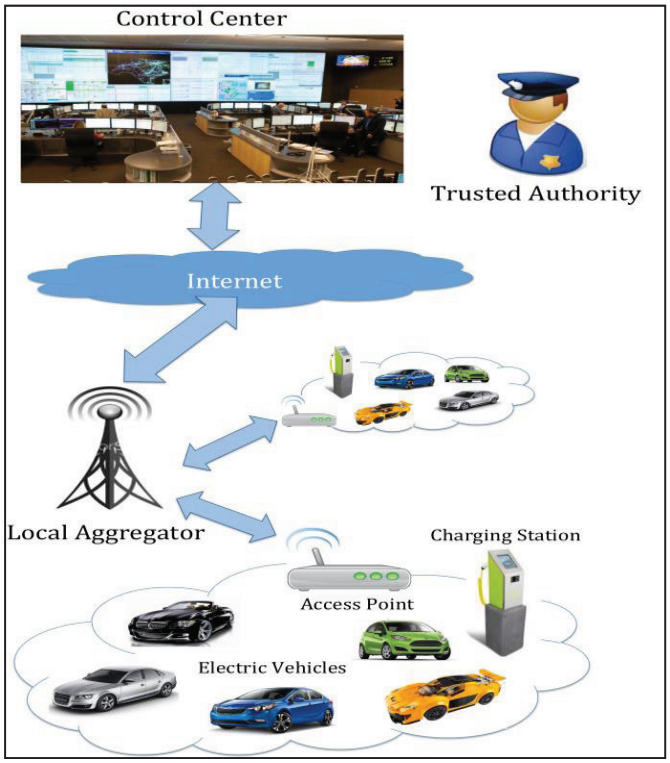



Fig. 1: System Model

*4) Signing:* We utilize Rabin signature algorithm [31], where $H$ is a collision-resistant hash function $H : \{0,1\}^* \rightarrow \{0,1\}^k$, The public key is $n$, and the private key is the pair $(p, q)$. To sign a message $m$, pick random padding $U$ and calculate $H(mU)$ so that $H(mU)$ is a square modulo $n$, $x^2 = H(mU)\ mod\ n$. The signature on $m$ is the pair $(U, x)$.

*5) Verification:* Given $m$ and $(U, x)$, calculate $x^2$ and $H(mU)$. Then verify if $x^2 = H(mU)\ mod\ n$.

BlueJay is a combination of Hummingbird-2 lightweight authenticated encryption scheme and PASSERINE cryptosystem optimized for a 1024-bit public modulus $n$ and 32-bit register size.

The proposed scheme also utilizes AKARI-2 [32], a lightweight pseudorandom number generator, to generate pseudonym IDs (PIDs) and symmetric keys. AKARI-2 is simple and easy to implement; it is appropriate for limited-computation devices, e.g., radio frequency identification, and sensors.

## V. The Proposed Scheme

Our proposed scheme has three phases: the first one is the initialization phase, which setups the system components and generates the keys and security parameters; the second phase is the operation phase, which is responsible for electricity-trade operation; and the final phase is the billing phase to pay the electricity expenses.

### A. Initialization Phase

The initialization phase establishes the network and defines the required security parameters.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TVT.2016.2577018, IEEE Transactions on Vehicular Technology



First, TA issues two pairs of public-private keys to both CC and LAs as follows:

- For CC, TA generates CC's private key pair $\{p_{cc}, q_{cc}\}$ and corresponding public key $n_{cc}(n_{cc} = p_{cc}\, q_{cc})$.

$$TA \xrightarrow{n_{cc},\, p_{cc},\, q_{cc}} CC.$$

- For each LA, TA generates LA's private key pair $\{p_{la}, q_{la}\}$ and public key $n_{la}(n_{la} = p_{la}\, q_{la})$.

$$TA \xrightarrow{n_{la},\, p_{la},\, q_{la}} LA.$$

- Also, TA assigns a secret ID and secret session key for each CS; the station includes its secret ID in its messages to LA to proof its identity. Thus, this ID should be stored in a secure memory module.

$$TA \xrightarrow{\mathcal{L}_{cs_j}, k_{cs_j}} CS_j.$$

- Each EV generates a PID utilizing AKARI-2, $EV_i = AKARI(x0, x1)$, where $x0, x1$ are initial seeds. Each EV changes its ID from time to time for more anonymity, e.g., a different PID for each connection session.

- In addition, EVs generate symmetric keys to secure the connection sessions with LA. $k_i = AKARI(h0, h1)$, where $h0, h1$ are initial seeds. This key changes frequently so that the probability of compromising it is diminished; then, messages' confidentiality is enhanced.

### *B. Operation Phase*

The operation phase is responsible for the electricity trade procedure; it organizes the tasks between different participants during the power purchase. First, we need to define a type of messages called *request messages*, which are utilized to manage the electricity exchange process. The request message is a request to buy/sell electrical power; it consists of the required amount of electricity and the current price. The request message could be sent from EVs to LA to (dis)charge their batteries or from CC to LA to supply electricity to EVs or consume electricity from them. There are four request messages: the first one is *the CC supply request message*, which is a request from CC to sell a share of electricity. The second type is *the CC consume request message*, which is a request from CC to purchase an amount of electricity. The third message is *the EV charge request message*, which is a request from an EV to charge its battery, i.e., the EV wants to buy a portion of power from the grid. Last request message is *the EV discharge request message*, which is a request from an EV to discharge its battery; that means the EV wants to sell a portion of its stored power. It can be inferred that there are four cases of electricity transfer operations:

*1) Case 1. The CC Supply Request:* In this case, CC wants to sell a portion of grid's electricity to EVs. Consequently, CC should follow a specific procedure:

- CC first sends *the CC supply request message* to the connected LAs; the message $M_s$ should indicate the amount of sold electricity $A$ and the price that the grid asked for $x$ (the price is in form $x$ cents/KWH), $M_s = (A, x)$.

- CC only signs the message by its private key to authenticate it, as the contents of that message are not confidential. First, a timestamp and a nonce are involved to the message to prevent replay attacks, $m_s = (M_s \,|T_{s1}\, |L_1)$. Then, CC picks a random padding $U_1$ and calculates $H(m_s U_1)$ so that $H(m_s U_1)$ is a square modulo $n_{cc}$, $x_1^2 = H(m_s\, U_1)\, mod\, n_{cc}$. The signature on $m_s$ is the pair $(U_1, x_1)$. Consequently, CC sends the signed message $(m_s, U_1, x_1)$ to the connected LAs.

- When LA receives *the CC supply request message*, it checks the validity of CC's signature by recalculating $(x_1, U_1)$ and checks if $x_1^2 = H(m_s U_1)\ mod\ n_{cc}$ is hold. Then, LA verifies the timestamp $T_{s1}$ and nonce $L_1$ values. Subsequently, LA signs $M_s$ by its private key to authenticate it to its connected EVs. The message also involves a timestamp and a nonce, $m_{s1} = (M_s \,|T_{s2}\, |L_2)$. Using a random padding $U_2$, LA computes $x_2^2 = H(m_{s1} U_2) mod\ n_{la}$. Then, it forwards $(m_{s1}, U_2, x_2)$ to the existing EVs via APs.

- If an EV wants to participate in that session and charge electricity from the grid, it verifies the signature of LA by calculating $(U_2, x_2)$ and checks if the $x_2^2 = H(m_{s1} U_2) mod n_{la}$ is hold and then verify the validity of timestamp $T_{s2}$ and nonce $L_2$. Then, it checks the message's contents. While if it does not want to charge its battery, it simply discards/ignores the message.

- If one or more EVs are satisfied by the current price and want to purchase the offered electricity from CC, they response to LA by *the EV charge request message*, which contains the desired amount of electricity to buy. When EV responses to LA, it means an implicit acceptance for the offered price; only EVs that are interested to buy by the offered price will reply while the remaining EVs do not reply. EVs encrypt their charge requests by LA's public key. In addition to the required amount of power, the messages include EV's PID and a secret session key. If the EV, with current PID $EV_i$, wants to purchase $W$ amount of power by the price $X$, then it generates a secret key $k_i$. Next, EV combines the required amount $W$, its current PID $EV_i$, and the secret key $k_i$ and includes timestamp $T_{s3}$ and nonce $L_3$ values, $M_c = W \,|EV_i\, |k_i\, |T_{s3}\, |L_3$. Subsequently, it encrypts the result $M_c$ using $n_{la}$:

EV generates a set of co-prime numbers $b_1, b_2, \ldots, b_j$, and a random value $Y_1$ and then encrypts $M_c$ to obtain:

$$m_{c1} \equiv M_c^2 + Y_1\, n_{la} (mod\, b_0),$$
$$m_{c2} \equiv M_c^2 + Y_1\, n_{la} (mod\, b_1),$$
$$\ldots\ldots\ldots,$$
$$m_{cj} \equiv M_c^2 + Y_1\, n_{la} (mod\, b_j).$$
$$m_c = \{m_{c1}, m_{c2}, \ldots, m_{cj}\}.$$

Afterward, it sends $m_c$ to LA via the connected AP.

- If a particular EV is chosen for the current charging session, LA uses its suggested secret key to encrypt the messages to EV. LA sends the acceptance messages to the selected EVs encrypted by the previously shared session keys; each message includes the charge/discharge order (i.e., charging order in this case), the location of target CS, and the payment approach. For instance, if LA chooses the EV with current PID $EV_i$, then LA encrypts the acceptance message with its session key $k_i$; $M_a = E_{k_i}(charge, L_{cs}, cash/token)$, where $L_{cs}$ is

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TVT.2016.2577018, IEEE Transactions on Vehicular Technology



the location of target CS, and $cash/token$ determines the payment way. The used symmetric encryption scheme is the lightweight Hummingbird-2 cryptosystem that further reduces the computation overhead.

- LA selects a number of EVs that satisfies the offered power from the grid. EV's selection depends on the number of responses and the location of replied EVs. If the amount of the electricity from responses is more than supply, LA selects number of EVs that satisfies the supply and nearer to the CSs.

- The payment approach determines the way that EVs should follow to pay the electricity expenses. EVs pay by either cash or coupons (tokens). The coupons are anonymous authenticated tokens that the grid gives to the EV as a payment for a previous discharging session. The token only shows its financial value without any information about its holder; for instance, $EV_1$ discharges 300 KW of electricity to the grid via $CS_3$, and the grid's operator pays to $EV_1$ the price, which equals 100 dollar, by a 100-dollar token. The token shows only its value, the 100 dollar, but no data about the process's details or involved parties is included.

- EV first pays the required price to LA via a payment message $M_p, M_p = E_{k_i}(cash/token)$. Then, LA sends a confirmation message to the assigned CS to charge EV's battery. The confirmation message $M_f$ contains the assigned power to that EV and its current PID. The message also involves CS's ID $\mathcal{L}_{cs}$ and timestamp and nonce values $m_f = (M_f \,|\mathcal{L}_{cs}\,|T_{s4}\,|L_4)$, then $m_f$ is encrypted by the pre-shared symmetric key between CS and LA $k_{cs}, MF = E_{k_{cs}}(m_f)$. After CS checks the message's validity, it charges the EV by the approved amount of electricity. In other words, EV pays the declared price to LA first before receiving any electricity.

At the end of the current supply round, LA only stores the total amount of supplied electricity $A$ and the corresponding total profit for this round $X$; it keeps this information in its database to calculate the total bill for the whole month later. At the same time, CC saves the same data in its record; this step conserves the accountability and guarantees the correctness of LA's total bill. As LA calculates the total bill at the end of the month and sends it to CC, which checks its correctness by comparing it with the corresponding value in its record. Fig 2 shows the CC Supply Request Case.

*2) Case 2. The CC Consume Request:* When CC wants to purchase a portion of electricity from the existing EVs, it should follow a specific procedure:

- CC first sends *the CC consume request message* to the connected LAs; the message $M_n$ should indicate the amount of required electricity $C$ and the price that the grid offers $y$ cents/KWH, $M_n = (C, y)$. CC only signs the message to authenticate it. The message also involves a timestamp and a nonce to prevent replay attacks, $m_n = M_n\,|T_{s5}\,|L_5$. Then, CC picks a random padding $U_4$ and calculates $H(m_nU_4)$ so that $H(m_nU_4)$ is a square modulo $n_{cc}$, $x_4^2 = H(m_nU_4) mod n_{cc}$. The signature on $m_n$ is the pair $(U_4, x_4)$. Therefore, CC sends the signed message $(m_n, U_4, x_4)$ to LAs.

- When LA receives *the CC consume request message*, it checks the validity of the signature by recalculating the values $(U_4, x_4)$, checks if the $x_4^2 = H(m_nU_4) mod n_{cc}$ is hold, and then verifies timestamp $T_{s5}$ and nonce $k_5$ values.

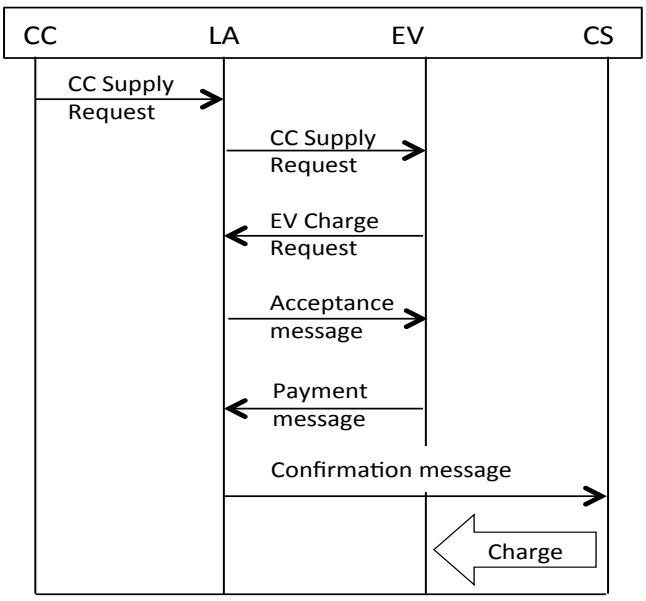



Fig. 2: CC Supply Request Case

Next, LA signs the message to authenticate it to EVs. The message also involves a timestamp $T_{s6}$ and nonce $k_6$ to prevent replay attacks, $m_{n1} = (M_n\,|T_{s6}\,|K_6)$, and then LA signs the message as: using the random padding $U_5$, LA computes $x_5^2 = H(m_{n1}U_5) mod n_{la}$. Then, it forwards the signed message $(m_{n1}, U_5, x_5)$ to the existing EVs.

- If one or more EVs want to sell their stored electricity to CC, they first verify the signature of LA by computing $(U_5, x_5)$, checks if the $x_5^2 = H(m_{n1}U_5) mod n_{la}$ is hold, and then verifies timestamp $T_{s6}$ and nonce $k_6$ values. While the other uninterested EVs ignore the request message from the first place.

- The interested EVs then response to LA by *the EV discharge request message*, which contains the desired amount of power to sell. EVs encrypt their discharge requests by LA's public key. In addition to the amount of sold power, the messages include the EV's PID and the secret session key. If the EV, which currently has the PID $EV_r$, wants to sell $D$ amount of power to the grid by the price $y$, then it generates a secret key $k_r$. Next, the EV combines the required amount $D$, its current PID $EV_r$, and the secret key $k_r$, includes the timestamp $T_{s7}$ and nonce $K_7$ values, $M_d = D\,|EV_r\,|k_r\,|T_{s7}\,|K_7$, and then encrypts the result $M_d$ using $n_{la}$:

EV generates a set of co-prime numbers $a_1, a_2, \ldots, a_e$ and a random value $Y_2$ and then encrypts $M_d$ by LA's public key as follows:

$$m_{d1} \equiv M_d^2 + Y_2\, n_{la} (mod\, a_0),$$
$$m_{d2} \equiv M_d^2 + Y_2\, n_{la} (mod\, a_1),$$
$$\ldots\ldots\ldots,$$
$$m_{de} \equiv M_d^2 + Y_2\, n_{la} (mod\, a_e).$$
$$m_d = \{m_{d1}, m_{d2}, \ldots, m_{de}\}.$$

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TVT.2016.2577018, IEEE Transactions on Vehicular Technology



Afterward, EV sends the encrypted message $m_d$ to LA.

- If a particular EV is chosen for the current discharging session, LA uses its suggested secret key to encrypt the messages to the selected EV. For instance, if LA chooses the EV with current PID $EV_r$, then LA encrypts the acceptance message with the session key $k_r$; the message LA's acceptance message to EV contains the discharge order, the CS's location $L_{cs}$, and the payment order, $M_{a1} = E_{k_r}(discharge, L_{cs}, cash/token)$.
- EV discharges the agreed amount of electricity to the assigned station first, and then CS sends a confirmation message $M_{f1}$ to LA. CS also involves its ID $\mathcal{L}_{cs}$ and timestamp and nonce values, $m_{f1} = (M_{f1} \,|\mathcal{L}_{cs} \,|T_{s8} \,|K_8)$, before encrypts the message using the pre-shared symmetric key $k_{cs1}$, i.e., $MF_1 = E_{k_{cs1}}(m_{f1})$. Then, LA pays the price of sold power to EV at the moment via a payment message $M_{p1}$ (e.g., $M_{p1} = E_{k_r}(token)$ is a token payment message to $EV_r$). In other words, EV first transfers the contracted electricity to the grid, and then LA directly pays the obligated price to it.

At the end of the current consume round, LA, CC as well, only stores the total amount of sold electricity $C$ and the corresponding total price for the round $Y$. Fig 3 shows the CC Consume Request Case.

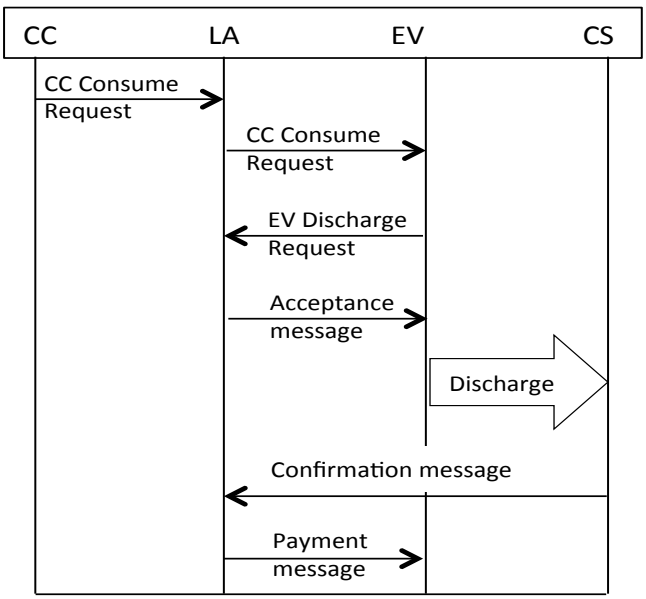



Fig. 3: CC Consume Request Case

*3) Case 3. The EV Charge Request:* In this case, EV asks for electricity from the grid, i.e., wants to purchase a portion of electricity from the grid. So, EV follows a specific procedure:

- EV first sends *the EV charge request message* to LA; the message should indicate the amount of required electricity $S$ and the price $z$ that EV can afford. EV encrypts the message by LA's public key; it attaches its current PID and a secret session key to the request. The message also involves a timestamp and a nonce to prevent replay attacks. If the EV, which currently has the PID $EV_q$, wants to sell $E$ amount of power to the grid by the price $w$, then it generates a secret key $k_q$. Next, EV combines the required amount $E$, its current PID $EV_q$, and the secret key $k_q$, includes timestamp $T_{s9}$ and nonce $K_9$ values, $M_{cr} = E \,|EV_q \,|k_q \,|T_{s9} \,|K_9$, and then encrypts the result $M_{cr}$ using the connected LA's public key $n_{la}$:

EV generates a set of co-prime numbers $d_1, d_2, \ldots, d_v$ and a random value $Y_3$ and then encrypts the $M_{cr}$ by LA's public key to obtain a series of cipher texts:

$$m_{cr1} \equiv M_{cr}^2 + Y_3\, n_{la} (mod\, b_0),$$
$$m_{cr2} \equiv M_{cr}^2 + Y_3\, n_{la} (mod\, b_1),$$
$$\ldots\ldots\ldots,$$
$$m_{crv} \equiv M_{cr}^2 + Y_3\, n_{la} (mod\, b_j).$$
$$m_{cr} = \{m_{cr1}, m_{cr2}, \ldots, m_{crv}\}.$$

Afterward, EV sends $m_{cr}$ to LA.

- When LA receives *the EV charge request message*, it decrypts the message and verifies the attached timestamp and nonce values. Then, LA aggregates the total amount of requested electricity from all EVs in the connected clusters. $Q = \sum_{EV_{charge}} S_{EV_{charge}}$, where $EV_{charge}$ is the total number of EVs that ask to charge their batteries in LA's connected clusters. LA then sends a request message by the total request to CC. The request also includes the different suggested prices by EVs. The message should involve timestamp $T_{s10}$ and nonce $K_{10}$ to prevent replay attacks; $m_t = Q \,|T_{s10} \,|K_{10}$. Then, it is signed by LA's private key and encrypted by CC's public key. LA picks a random padding $U_6$ and calculates $H(m_n U_6)$ so that $H(m_n U_6)$ is a square modulo $n_{la}$, $x_6^2 = H(m_n U_6) mod n_{la}$. The signature on $m_t$ is the pair $(U_6, x_6)$. Therefore, the signed message $M_T = (m_t, U_6, x_6)$ is encrypted by $n_{cc}$.

LA generates a set of co-prime numbers $c_1, c_2, \ldots, c_h$ and a random value $Y_4$ and then encrypts the $M_T$ by CC's public key to obtain a series of cipher texts:

$$m_{T1} \equiv M_T^2 + Y_4\, n_{cc} (mod\, c_0),$$
$$m_{T2} \equiv M_T^2 + Y_4\, n_{cc} (mod\, c_1),$$
$$\ldots\ldots\ldots,$$
$$m_{Th} \equiv M_T^2 + Y_4\, n_{cc} (mod\, c_h).$$
$$m_T = \{m_{T1}, m_{T2}, \ldots, m_{Th}\}.$$

Afterward, LA sends the message $m_T$ to CC.

- CC decrypts the message and verifies the signature of LA and then checks validity of timestamp and nonce, then it checks if its resources can cover the required amount of electricity. It also verifies the offered prices from different EVs and set the final price for its sold power. CC then sends a confirmation message to each LA; this message contains the final price and the location of CSs for each region. This message is signed by CC's private key and then encrypted using LA's public key.

- LA sends an order message to each EV; the order message contains the location of target CS and the corresponding price. LA encrypts the order message to EV by the previously shared session key; also, the message includes EV's PID. For example, if LA chooses the EV with current PID $EV_q$, then LA encrypts the order message with its shared secret key

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TVT.2016.2577018, IEEE Transactions on Vehicular Technology



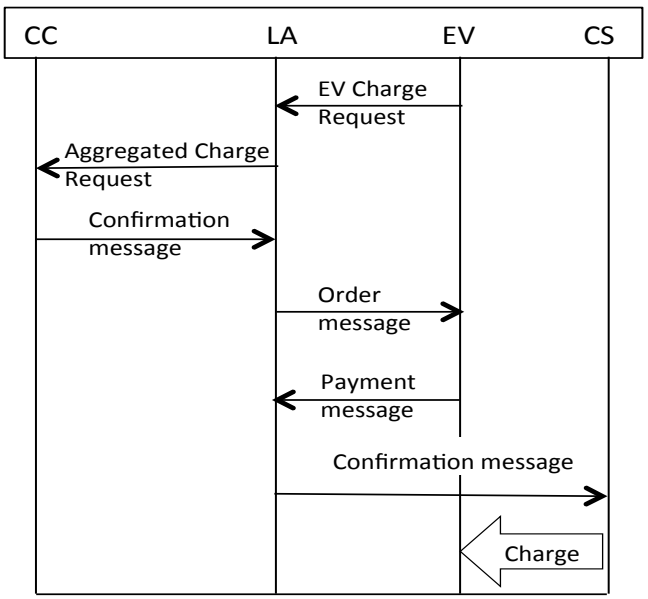



Fig. 4: EV Charge Request Case

$k_q$; $M_o = E_{k_q}(charge, L_{cs}, cash/token)$, where $L_{cs}$ is the location of target CS.

- EV first pays the required price to LA; it sends the price in a payment message, e.g., $EV_q$ pays to LA by $M_{p2} = E_{k_q}(cash/token)$. Next, LA sends a confirmation message to CS. The confirmation message $M_{f2}$ contains the assigned power to that EV and its current PID $EV_q$. The message involves a timestamp and nonce values $m_{f2} = (M_{f2} \mid \mathcal{L}_{cs} \mid T_{s11} \mid K_{11})$. LA then encrypts $m_{f2}$ by the pre-shared secret key between CS and LA $k_{cs}$, $MF_1 = E_{k_{cs}}(m_{f2})$. Consequently, CS charges the agreed amount of electricity to EV after checking the message's validity. In other words, EV pays to LA before charging its battery.

At the end of the current charging round, LA only stores the total amount of sold electricity $Q$ and the corresponding total profits $H$ for this round. LA keeps this information to calculate the total bill for the month. CC does the same calculations and saves the results in its record too. Fig 4 shows the EV Charge Request Case.

*4) Case 4. The EV Discharge Request:* In this case, EV wants to discharge its battery to the grid, i.e., sell a portion of its electricity to the grid. Then, EV follows a specific procedure:

- EV first sends *the EV discharge request message* to LA; the message should indicate the amount of sold electricity $Wt$ and the price that EV offers $l$. EV encrypts the message by $n_{la}$; it attaches to the request its current PID and the suggested secret session key. The message also includes a timestamp and a nonce to prevent replay attacks.

- LA aggregates the total offered electricity amount from the interesting EVs and sends a request message by the total amount to CC. The message is signed by LA's private key and encrypted by CC's public key. CC computes the current needs of the grid and estimates the suitable corresponding price. It then sends confirmation messages to LAs; the confirmation message contains the electricity amount, the price, and the location of CSs in LA's region.

- Next, LA sends order messages to EVs that are interested to discharge their batteries to the grid; each order message contains the location of target CS and the price for discharging power. LA encrypts the order message by the previously shared session key with that EV.

- EV first discharges the agreed amount of electricity to the required CS; CS then sends a confirmation message to LA. Then, LA pays to EV the price of sold power at the moment via a payment message. In other words, EV discharges the assigned electricity first, and then LA directly pays the corresponding price to it.

At the end of the current discharge round, LA (also CC) only stores the total amount of sold electricity $Qc$ and the corresponding total price $L$. Fig 5 shows the EV Discharge Request Case.

### C. Billing phase

First, both of LA and CC compute the total amount of exchanged power and the corresponding price or profit for different cases for the whole month. For *the CC Supply Request Case*, the total sold electricity is $A_{all} = \sum_{b_s} A_{b_s}$ and the corresponding profit $X_{all} = \sum_{b_s} X_{b_s}$, where $b_s$ is the total number of the CC Supply Request sessions per month. According to *the CC Consume Request case*, the total purchased electricity $C_{all} = \sum_{b_c} C_{b_c}$ and its total price $Y_{all} = \sum_{b_c} Y_{b_c}$, where $b_c$ is the total number of the CC Consume Request rounds for the month. In *the EV Charge Request case*, the total amount of charged electricity equals $Q_{all} = \sum_{b_h} Q_{b_h}$ and the corresponding profit $H_{all} = \sum_{b_h} H_{b_h}$, where $b_h$ is the total number of the EV Charge Request sessions per month. For *the EV Discharge Request case*, the total amount of discharged electricity equals $Qc_{all} = \sum_{b_d} Qc_{b_d}$ and the corresponding profit $L_{all} = \sum_{b_d} L_{b_d}$, where $b_d$ is the total number of the EV Discharge Request sessions per month. Next, they compute the total sold electricity $E_S = A_{all} + Q_{all}$ and the corresponding profit $P_F = X_{all} + H_{all}$, and the total purchased electricity $E_U = C_{all} + Qc_{all}$ and its total price $P_R = y_{all} + L_{all}$. Then, LA signs the billing information $m_B = (E_S, P_F, E_U, P_R)$ by its private key, i.e., after adding timestamp and random nonce values, and then encrypts it by CC's public key. Next, LA sends the resulted bill message $B$ to CC.

$$LA \xrightarrow{B} CC.$$

After decrypting the message and checking LA's signature, CC compares LA message with its computed information. Subsequently, CC computes the net price and pays to LA via a payment message $M_P$, which is signed by $(p_{cc}, q_{cc})$ and encrypted by $n_{la}$.

$$CC \xrightarrow{M_P} LA.$$

It is obvious that EVs deal only with the related LA and have no connection with CC. Moreover, LA communicates with EVs via their PIDs, which are changing frequently. EV may use different PID for each communication session with LA. Furthermore, EV should take action first in all cases, because it is the untrusted party. When CC needs to supply

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TVT.2016.2577018, IEEE Transactions on Vehicular Technology



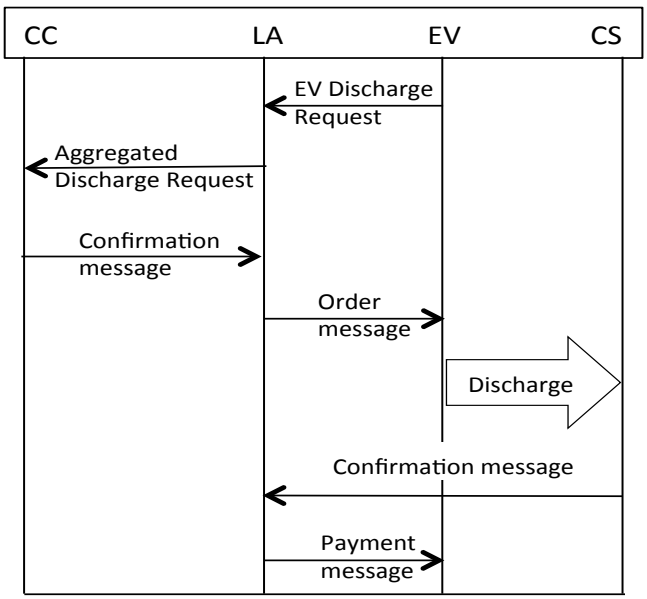



Fig. 5: EV Discharge Request Case

power or EV wants to discharge its battery, EV transmits the electricity to grid first, and then receives the payment from LA. Accordingly, EV should pay the electricity price first before charging its battery. In addition, LA does not even need to save the various PIDs for different connected EVs; it only saves the total amount of traded power and the corresponding total price/profit for each round, e.g., if LA connects with $50$ EVs in a CC supply request case, it stores the total power sold to the $50$ EVs and the corresponding total profit. As a result, the privacy of EVs is preserved; EV's real ID, exact location and personal information are protected. At the same time, LAs and CC assure their profits and overcomes the trust problem, i.e., avoid malicious EVs. The scheme also ensures the traceability of electricity trade operation.

## VI. Security Analysis

The proposed scheme attempts to guarantee several security requirements for V2G connections simultaneously: EVs' privacy (i.e., owner identity and vehicle location), confidentiality and messages integrity, and overcomes EVs' authenticity problem and achieves the power trade's accountability.

*EVs' Privacy*. The proposed scheme guarantees EV's privacy by preserving EV's real identity from being exposed to any party, i.e., adversaries, aggregators, or even grid's operator. Our scheme allows each EV to generate its own PIDs; these PIDs can be changed per session or just from time to time. This frequent change assists in preserving EV's private information so that no party can link between the EV and a specific location or ID. For instance, $EV_i$ needs to charge its battery by the value $x$ at time $t_1$. So, $EV_i$ selects a pseudo identity $PID_1$ and new secret session key $k_1$ for this session. At time $t_2$, $EV_i$ wants to sell a certain electricity amount $y$ to the grid; then, it chooses another pseudo identity $PID_2$ and secret session key $k_2$ to that new session. Therefore, no party can link between these two sessions and $EV_i$, as $EV_i$ has finished all the related processes including payment during the session. In the charging session, it pays for $x$ amount, and then charges its battery. While in the discharging session, $EV_i$ injects $y$ to the grid before receiving the payment. No need to store information about different PIDs and session keys for $EV_i$ in LA or grid's records, as it is not useful for LA or CC after session's end. As a result, neither LA nor CC can link between $EV_i$, i.e., its real identity or pseudo ones, and these sessions.

In addition, if an adversary $\mathscr{A}$ could extract certain messages from the two sessions, he/she cannot discover that the involved EV in the two sessions is the same vehicle. In other words, $\mathscr{A}$ cannot link between the EV and its activities. Furthermore, grid's operator and associated LAs do not need to know the real identities of EVs; they guarantee grid's profit without tracing EVs' real identities. So, they will not ever ask for EV's real identity. Then, if an EV receives any request to expose its real identity, it knows that the requester is an adversary and blocks that malicious request. Suppose $\mathscr{A}$ compromises the legal $LA_i$ and contacts with its EVs. $\mathscr{A}$ may try to obtain information about EVs' real identities to seize certain financial gain. In this case, EVs know that $LA_i$ is compromised. So, they ignore its request and report to the grid's operator about that LA. The operator verifies LA's status after receiving a certain number of bad reports about it.

*Authenticity*. The signatures of CC and LA authenticate their messages. For example, the CC supply request message is authenticated by CC's signature; no malicious party can forge it. On the other hand, grid's operator guarantees that the involved EVs operate honestly; it forces them to take the action first to guarantee grid's financial profit. In addition, EV's current PID and shared key authenticate its messages during the running session. The proposed scheme forces EVs to follow a specific procedure to prevent them from acting maliciously, i.e., EV has no chance to misbehave, because it has to finish its part first, and then receive the corresponding action from the grid. Even if $\mathscr{A}$ impersonates an EV, he/she cannot seize its benefits, because $\mathscr{A}$ has to follow the procedure by accomplishing his/her part in the electricity exchange operation first. Consequently, $\mathscr{A}$ is forced to behave honest to receive the electricity or the price. In summary, the scheme overcome EVs' authentication problem and guarantees the grid's financial profit.

*Confidentiality and Messages Integrity*. The proposed scheme assures confidentiality and integrity of the exchanged messages using a combination of public key and symmetric key schemes. In the CC Supply Request case, for instance, before CC sends the CC supply request message $M_s$, CC authenticates it by its signature. No party can modify its contents, but they can access it. In addition, no replay attacks can success, as the message contains timestamp $T_{si}$ and nonce $L_i$. Similarly, when LA forwards that message to EVs. When interested EVs response to LA by a charge request messages $M_c$. They encrypt it by $n_{la}$. So, confidentiality and integrity of their messages are guaranteed. For the acceptance message $M_a$, LA encrypts it by the secret key suggested by EV. So, this messages is secured against confidentiality or integrity attacks. For instance, if LA chooses EV with $PID_i$, and encrypts $M_a$ by its session key $k_i$, then any $\mathscr{A}$ cannot extract information, such as the target CS's identity, from the message. As well,

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TVT.2016.2577018, IEEE Transactions on Vehicular Technology



the confirmation message $M_f$ is encrypted by CS's secret key so that its confidentiality and integrity is assured too. In addition, the used secret keys are frequently changed so that the probability to compromise them is limited.

The proposed scheme utilizes the efficient BlueJay ultra-lightweight cryptosystem, which is a combination of lightweight public key PASSERINE scheme and Hummingbird-2 lightweight symmetric encryption scheme.

The lightweight public key PASSERINE scheme that used to encrypt the exchanged messages during the establishment of the secure sessions between the LA and EVs, e.g., EV (dis)charging request messages, is an enhanced lightweight version of Rabin cryptosystm. It has been practically proven that Rabin-based cryptosystems hardness problem is equivalent to the integer factorization problem.

**Theorem 1.** *Let $N = pq$, where $p \equiv q \equiv 3(mod\ 4)$ are primes, and define $S_{N,l} = \{1 \leq x < N : gcd(x, N) = 1, 2^l \mid (x + 1)\}$, where $x \in \mathbb{Z}_N^* - S_{N,l}$, then the probability that there exists $y \in S_{N,l}$ with $x \neq y$ but $x^2 \equiv y^2(modN)$ equals $1/2^{l-1}$. Then breaking the one-wanness security property of Rabin-based cryptosystems (i.e., factoring $N$ into $p$ and $q$) is in polynomial time if the redundancy bits in the ciphertext is $l = O(log(log(N)))$.*

*Proof.* Let $\mathcal{O}$ be an oracle that takes the public key $N$ and a ciphertext message $c$ (with $l$ redundant bits) as input and returns either the corresponding plaintext $m$ or an invalid value $Null$.

Choose a random $x \in \mathbb{Z}_N^*$ such that neither $x$ nor $N - x$ satisfy the redundancy scheme (i.e., the $l$ least significant bits are not all 1) and set the ciphertext $c = x^2(mod\ N)$ as input for $\mathcal{O}$. According to theorem's assumption, if the probability that one of the two unknown square roots of $c\ mod\ N$ has the correct $l$ least significant bits equals $1/2^{l-1}$, then $\mathcal{O}$ can output the plaintext $m$ from $c$, where $\acute{x} = 2^l m + (2^l - 1)$, then we have $(\acute{x})^2 \equiv x^2(mod\ N)$ and $\acute{x} \not\equiv \pm x(mod\ N)$. Hence $gcd(\acute{x} - x, N)$ will split $N$. If $l = O(log(log(N)))$, then it is required approximately $2^{l-1}$ trials to factor $N$ in polynomial time. As a result, factoring $N$ is NP-hard problem when $l \neq O(log(log(N)))$. □

While the lightweight symmetric encryption scheme Hummingbird-2, with an innovative hybrid structure of block cipher and stream cipher, is utilized to encrypt the exchanged messages during the (dis)charging sessions, such as acceptance or payment messages. Hummingbird-2 with its four optimal 4-bit S-Boxes belongs to a group of lightweight symmetric schemes that are proven to be resistible to differential, linear and algebraic attacks [33]. Moreover, the used secret keys to secure the exchanged messages during (dis)charging sessions are frequently changed, i.e., a new key for each session. For instance, $EV_t$ needs to charge its battery at time $t_1$. So, it selects a pseudo identity $PID_w$ and new secret session key $k_w$ for this session. At time $t_2$, $EV_t$ wants to sell a certain electricity to the grid so that it chooses another pseudo identity $PID_v$ and new secret session key $k_v$ to that new session. The key is used to encrypt only two messages during the session; consequently, the probability of compromising the session key is diminished.

*Accountability*. Some previous works attempt to achieve the accountability and tractability. They require saving the detailed information about previous sessions for each EV, and most of time, they need to reveal EV's real identity by the end of the connection. While our proposed scheme guarantees the operation's accountability without revealing any private information about EVs. EVs' real identities are concealed and never revealed to any party neither during connection sessions nor during the tracing operation. Also, grid's operator does not need to maintain all the detailed information about numerous previous power trade sessions for tracing purposes. Therefore, the proposed scheme saves the storage capacity of both LA and CC. They just need to keep the total amount of purchased/sold electricity and the corresponding price/profit for each session.

## VII. Performance Evaluation

In this section, we evaluate the performance of the proposed scheme in terms of communication overhead and computation complexity.

### A. Communication Complexity

The proposed scheme assures security and privacy demands for different parties in V2G connection with low communication overhead. During the initialization phase, TA sends only three messages to the parties: CC, LAs, and CSs. This process considers a trivial communication load for the high-capabilities TA. In addition, EVs, i.e., the restricted-capabilities parties, do not participate in this phase. Then, the communication overhead for that phase can be neglected.

According to the operation phase, the number of exchanged messages per session does not exceed six messages if only one EV is participated. Only one or two messages are sent by EV, which remains a minor communication duty for it. In the CC supply request case, CC sends one message to the connected LAs, and they forward this message to the involved EVs. Next, LA and the interested EV exchange three messages. Typically, EV receives three messages from LA and replies by one message. Also, LA receives one message related to this EV from the assigned CS. This process is repeated for all participated EVs in the session. Suppose the number of participated EVs per session is $q$, then the total number of exchanged messages in that case equals $[2+(4*q)]$ messages. Other three cases have also the same total overhead. In summary, the total communication overhead for each case is $[2 + (4 * q)]$ messages most of them handled by CC and LAs. While, the maximum communication overhead for each EV is constant and equals sending one or two messages per session. According to LA, its overhead is linearly increased, as the number of involved EVs increased. Thus, the total communication load increases linearly with the increase in the number of selected EVs. However, the increase in the total communication load is bounded by the maximum number of EVs in the parking lot, which can be roughly determined. Fig 6 shows the communication load for each EV and the total overhead for each (dis)charging session. As shown, the communication overhead for EV is constant and very low. Although the total communication burden, most of it is

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TVT.2016.2577018, IEEE Transactions on Vehicular Technology



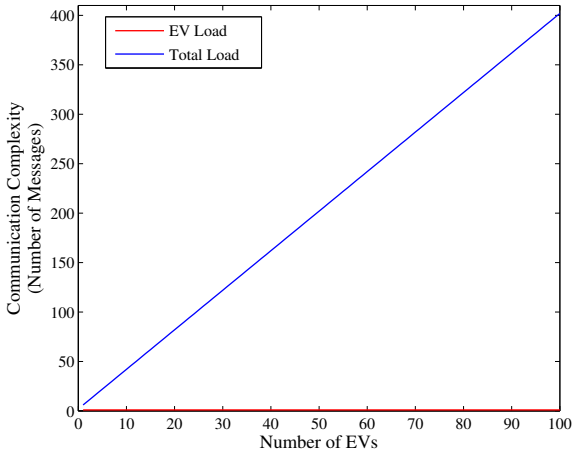



Fig. 6: Communication Complexity per Session

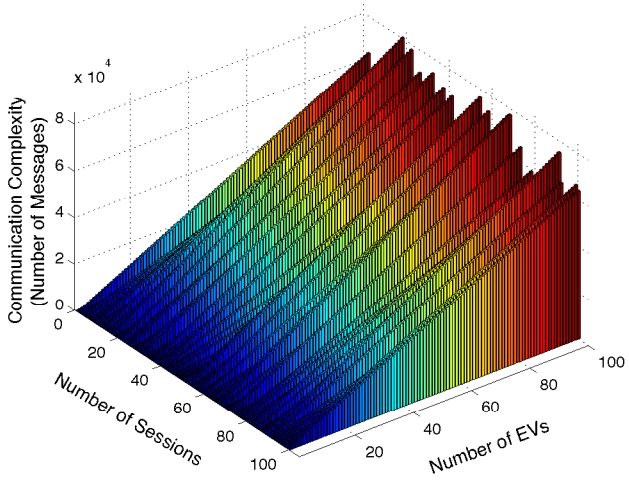



Fig. 7: Total Communication Complexity

handled by LA, is linearly increasing with the increase in EVs' number, it is still lightweight and bearable by the connection because of the limited number of participated EVs per session.

The total communication complexity for (dis)charging sessions per month equals $N = ([b_s+b_c+b_h+b_d]*[2+(4*q)])$. In billing phase, CC and LA exchange two messages. Then, the total communication overhead per month equals $([b_s+b_c+b_h+b_d]*[2+(4*q)])+2$ messages. Fig 7 illustrates the impact of EVs' number, also the effect of different number of electricity sessions on the total communication overhead per month. The overhead is increased by the increase of participated EVs' number. Also, the load is increased, if large number of sessions is performed during the month. However, the increase in communication complexity is limited and tolerable by different parties. As a result, our proposed scheme requires lightweight communication duty and suits limited-capabilities EVs.

In addition to guarantee the security requirements for V2G connection, such as EVs' owners and location privacy, the electricity trade information confidentiality and integrity, and the involved parties authentication, our proposed scheme offers more lightweight communication and computation overhead.

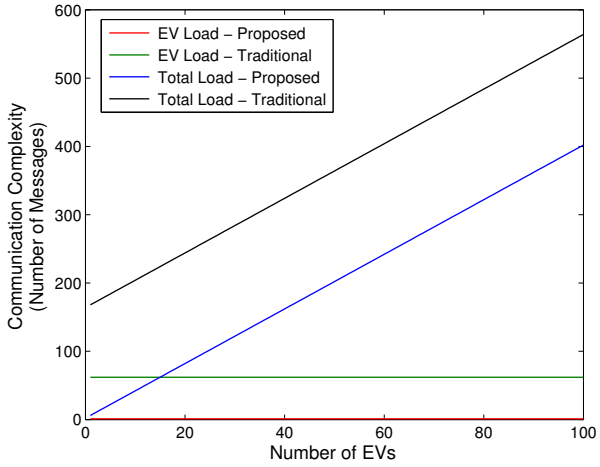



Fig. 8: Communication Complexity per Session for Proposed Scheme .vs. Traditional Scheme

We compare our proposed scheme with the scheme [20], which is an authentication scheme that also preserves EVs' real identities. Before the electricity (dis)charging operations begin, the central authority (CA), which is a third trusted party, issues a one-day permit for each EV so that EV can authenticate itself to LA by this permit while use a PID to preserve its privacy. EV and LA then exchange several messages to authenticate EV and create the required symmetric key for that session. Each EV is obligated to send periodic reports about its status to the connected LA during the session, i.e., we assume that each session lasts for one hour and EV sends its periodic report every minute. At the end of the (dis)charging session, LA sends a signed reward message to EV, which uses it to claim its profit from CA. On the other hand, our scheme does not require the different parties to have pre-shared secret parameters, also it does not need the presence of the third party during the operation phase. In addition, our scheme requires to exchange only six messages per session. Fig 8 shows the comparison between one session for the CC supply request case in our proposed scheme versus one charging session for [20]. The total number of exchanged messages during the charging session for [20] equals $(164 + 4 * q)$ messages, while the total exchanged messages' number in the CC supply request cases equals $(2 + 4 * q)$ messages. It can be seen that our proposed scheme's case tend to have less communication load than [20] case especially for the load on EVs. As a result, the proposed scheme saves the communication overhead for EVs and LAs; specifically when the number of connected EVs increased. In conclusion, our proposed scheme requires lightweight communication duty and suits the limited-capabilities EVs.

### *B. Computation complexity*

EVs require lightweight cryptosystems because of their limited-computation capabilities. The proposed scheme guarantees that feature by reducing the number of exchanged messages, i.e., limited number of messages to be encrypted, in addition to use a combination of lightweight public key and

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TVT.2016.2577018, IEEE Transactions on Vehicular Technology



symmetric key schemes. Assume that $T_s, T_v, T_{Ep}, T_{Dp}, T_{Es}$, and $T_{Ds}$ are signing, verification, public key encryption, public key decryption, symmetric key encryption, and symmetric key decryption computation times in ms, respectively. During the initialization phase, TA provides the secret parameters to CC and LAs, while EVs are not involved in that operation. They only require to generate PIDs using the tiny-overhead AKRI-2 scheme. Thus, the computation overhead for that phase can be neglected.

According to the operation phase, CC, LAs, and EVs should perform few crypto operations: certain processes are public key, and the remaining are symmetric key operations. Only three messages per session are encrypted using public key cryptosystem, while the remaining messages are ciphered by symmetric key system (i.e., the used public key system, Passerine, is a lightweight cryptosystem. In addition, symmetric key cryptosystems are much lightweight than public key systems). EVs are only burdened by a little share of computation load. In the CC supply request case, CC sends the CC supply request message to the connected LAs; this message is signed using its private key. LA first verifies CC's signature and signs the message before forwarding it to the connected EVs. Next, the interested EV encrypts a charge request message by $n_{la}$ and replies to LA. So, LA has to perform a public key decryption process for that message. After that, the shared symmetric key is used to encrypt the remaining messages, i.e. three messages, for that session. Typically, EV receives three messages from LA and replies by one message. Consequently, the total computation time for the CC supply request case per EV is $T_{supply} = [(2*T_s)+(2*T_v)+(1*T_{Ep})+(1*T_{Dp})+(3*T_{Es})+(3*T_{Ds})]$ ms. The EV's share in that load is $T_{EV_{supply}} = [(1*T_v)+(1*T_{Ep})+(1*T_{Es})+(1*T_{Ds})]$ ms. While, the total computation time for the CC supply request case for $q$ participated EVs is $T_{supply_{all}} = [(2*T_s)+[(q+1)*T_v]+(q*T_{Ep})+(q*T_{Dp})+(3*q*T_{Es})+(3*q*T_{Ds})]$ ms. The CC consume request case has the same total computation overhead per EV as the supply case $T_{consume} = T_{supply} = [(2*T_s)+(2*T_v)+(1*T_{Ep})+(1*T_{Dp})+(3*T_{Es})+(3*T_{Ds})]$ ms. While, the EV's share in that load is different $T_{EV_{consume}} = [(1*T_v)+(1*T_{Ep})+(2*T_{Ds})]$ ms. However, the total computation time for the CC consume request case for $q$ participated EVs is the same as $T_{supply_{all}}$; $T_{consume_{all}} = [(2*T_s)+[(q+1)*T_v]+(q*T_{Ep})+(q*T_{Dp})+(3*q*T_{Es})+(3*q*T_{Ds})]$ ms.

According to the EV charge request case, EV first sends the EV charge request message to LA; this message is encrypted by LA's public key. LA decrypts it, and then sends a request message by the total request from all EVs that are interested to purchase electricity from CC. The message is signed by LA's private key and encrypted by CC's public key. CC decrypts the message, verifies the signature of LA, and then sends a confirmation to LA; the confirmation message is signed by CC and encrypted using LA's public key. LA sends an order message to each EV, which is encrypted by the previously shared session key. EV then pays the assigned price to LA by a payment message, which is encrypted by the given symmetric key too. LA then sends a confirmation message to the assigned CS; that message is encrypted by the pre-shared key between LA and CS. In this case, the total number of crypto operations per EV is two signing and two verification processes in addition to three encryption and three decryption public key processes and three encryption and three decryption symmetric key processes. $T_{charge} = [(2*T_s)+(2*T_v)+(3*T_{Ep})+(3*T_{Dp})+(3*T_{Es})+(3*T_{Ds})]$ ms, where $T_{charge}$ is the total computation time for the EV charge request case per EV. The computation overhead for each charged EV $T_{EV_{charge}}$ is one public key encryption, one symmetric key encryption, and two symmetric key decryption processes. $T_{EV_{charge}} = [(1*T_{Ep})+(1*T_{Es})+(1*T_{Ds})]$ ms. While, the total computation time for the EV charge request case for $q$ EVs equals $T_{charge_{all}} = [(2*T_s)+(2*T_v)+[(q+2)*T_{Ep}]+[(q+2)*T_{Dp}]+(3*q*T_{Es})+(3*q*T_{Ds})]$ ms. Finally, in the EV discharge request case, the involved parties follow almost the same procedure as in the EV charge request case but for discharging EVs. So, the total number of operations per EV $T_{discharge}$ during that case is the same as the EV charge request case $T_{discharge} = T_{charge} = [(2*T_s)+(2*T_v)+(3*T_{Ep})+(3*T_{Dp})+(3*T_{Es})+(3*T_{Ds})]$ ms. The computation overhead of discharging EV $T_{EV_{discharge}}$ is a little different from $T_{EV_{charge}}$; it includes one public key encryption, and two symmetric key decryption processes. $T_{EV_{discharge}} = [(1*T_{Ep})+(2*T_{Ds})]$ ms. While, the total computation time for the EV discharge request case for $q$ EVs is the same as $T_{charge_{all}}$; $T_{discharge_{all}} = [(2*T_s)+(2*T_v)+[(q+2)*T_{Ep}]+[(q+2)*T_{Dp}]+(3*q*T_{Es})+(3*q*T_{Ds})]$ ms.

The total computation time for the operation phase per month equals $T_{op} = [(b_s*T_{supply_{all}})+(b_c*T_{consume_{all}})+(b_h*T_{charge_{all}})+(b_d*T_{discharge_{all}})]$. In billing phase, LA sends a billing message to CC, and CC replies by a payment message. Then, the computation load for billing phase is $T_{bill} = 2*[T_s+T_v+T_{Ep}+T_{Dp}]$. In summary, the total computation overhead per month equals $T = T_{op}+T_{bill} = ([b_s+b_c+b_h+b_d]*[(2*T_s)+[(q+1)*T_v]+(q*T_{Ep})+(q*T_{Dp})+(3*q*T_{Es})+(3*q*T_{Ds})])+(2*[T_s+T_v+T_{Ep}+T_{Dp}])$ ms. Most of the computation load is performed by LAs and CC, and this load is insignificant for their capabilities. While EVs' computation operations are mainly tiny symmetric crypto-operations.

The performance of our proposed scheme has been analyzed and evaluated deploying a hardware implementation of BlueJay cryptosystem with a $1024-bit$ public modulus $n$ and $32-bit$ register size. It runs on a Cortex M0 platform, i.e., simple and fast, cheap, low power, and smallest ARM processor, which is embed on different parties in the V2G connection. Fig 9 shows the computation time per EV and also the total computation time per (dis)charging session. Clearly, each EV performs a small fixed number of crypto operations; while, the remaining computation load are handled by LA. Although the computation load for LA is linearly increased by the increase in EVs' number, its load is bounded and manageable. Fig 10 shows the impact of different number of EVs and sessions on the total computation overhead per month. The total overhead is increased by the increase in EVs' number as well as the increase in the charging sessions' number. However, the increase in computation load is limited

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TVT.2016.2577018, IEEE Transactions on Vehicular Technology



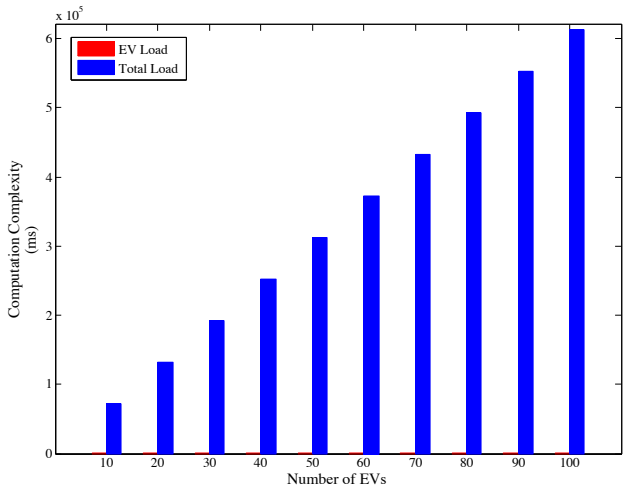



Fig. 9: Computation Complexity per Session

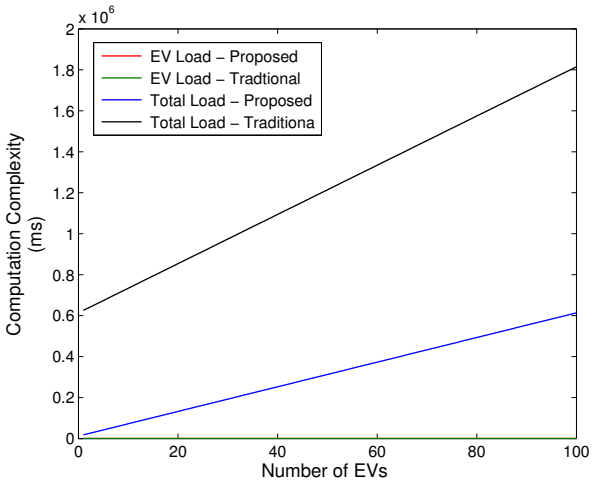



Fig. 11: Computation Complexity per Session for Proposed Scheme .vs. Traditional Scheme

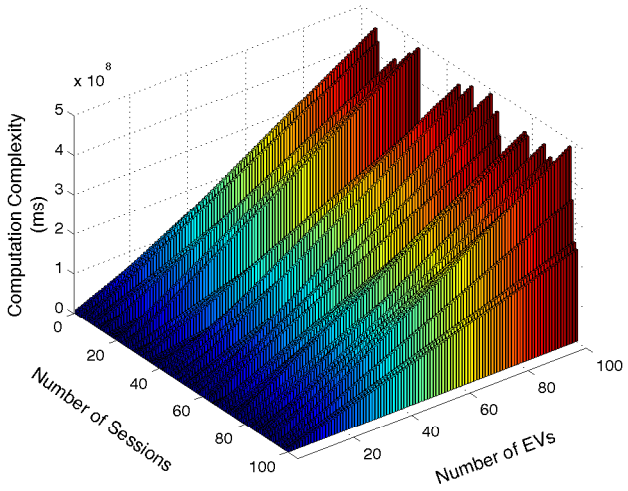



Fig. 10: Total Computation Complexity

and tolerable by different parties. The proposed scheme saves the computation time and preserves the processing abilities for the participated parties in the electricity trade operation especially EVs.

In Fig 11, we compare the proposed scheme with the scheme in [20]. It can be seen that there is a big gap between the total computation delay in two schemes for the benefit of our scheme. The total delay increases from 18023.03 to 613560.64 msec for the proposed scheme versus 625558.87 to 1815078.95 msec in [20] for the whole session. However, there is no huge difference in the computation delay per EV in both schemes; 15.18 msec for the proposed scheme versus 20.52 msec for the traditional one. Then, our scheme saves the total computation time and preserves the processing abilities for the participated parties in electricity trade operation.

In summary, the proposed scheme not only guarantees the security requirements for all involved parties, i.e., CC, LAs, and EVs, but also provides light computation and communication overhead.

## VIII. Conclusion

We have proposed a lightweight security and privacy-preserving scheme for V2G connection. The proposed scheme can guarantee several security requirements of V2G connections simultaneously. It preserves EV's owners and location privacy, diminishes the impact of malicious EVs, and overcomes EV's frequent authentication concern; it also assures confidentiality and integrity of the exchanged electricity trade messages. Moreover, the scheme keeps accountability and electricity-exchange operations tractability. Simulation results have demonstrated that the proposed scheme reduces the overall communication and computation overhead for V2G connection, as it decreases the number of exchanged messages between different parties; especially the messages sent by EVs. In addition, using a combination of symmetric key and lightweight public key schemes is further reducing the computation complexity. For the future wok, we will design the optimal selection technique for EVs during the electricity-trade process.

## References

[1] A. Abdallah and X. Shen, "Lightweight security and privacy-preserving V2G connection scheme in smart grid," *in Proc. IEEE Globecom'15*, USA, December 2015.

[2] X. Fang, S. Misra and D. Yang, "Smart grid – the new and improved power grid: A survey," *IEEE Communications Surveys & Tutorials*, vol. 14, no. 4, pp. 944 – 980, FOURTH QUARTER 2012.

[3] P. Chen, S. Cheng, "Smart attacks in smart grid communication networks," *IEEE Communications Magazine*, pp. 24 – 29, August 2012.

[4] J. Soares, T. Sousa and P. Faria, "An optimal scheduling problem in distribution networks considering V2G," *in Proc. IEEE CIASG*, France, April 2011.

[5] T. Yiyun, L. Can and L. Lin, "Impacts of electric vehicles on power grid," *in Proc. SUPERGEN*, China, September 2012.

[6] M. Galus, R. Waraich and G. Andersson, "Integrating power systems, transport systems and vehicle technology for electric mobility impact assessment and efficient control," *IEEE Transactions on Smart Grid*, vol. 3, no. 2, pp. 934 – 949, June 2012.

[7] T. Sousa, H. Morais and J. Soares, "Intelligent energy resource management considering vehicle-to-grid: A simulated annealing approach," *IEEE Transactions on Smart Grid*, vol. 3, no. 1, pp. 535 – 542, March 2012.

[8] M. Li, S. Salinas and P. Li, "LocaWard: A Security and Privacy Aware Location-Based Rewarding System," *IEEE Transaction on Parallel And Distributed Systems*, Vol. 25, No. 2, February 2014.

[9] W. Luo and U. Hengartner, "Veriplace: a privacy-aware location proof architecture," *In Proc. ACM SIGSPATIAL GIS*, USA, November 2010.

[10] C. Ardagna, M. Cremonini, S. De Capitani di Vimercati, and P. Samarati, "An obfuscation-based approach for protecting location privacy," *IEEE Transactions on Dependable and Secure Computing*, Vol. 8, No. 1, pp. 13 – 27, January/February 2011.

[11] A. Abdallah and X. Shen, "A lightweight lattice-based security and privacy-preserving scheme for smart grid," *in Proc. IEEE Globecom'14*, USA, December 2014.

[12] M. Mustafa, N. Zhang and Z. Fan, "Smart electric vehicle charging: Security analysis," *in Proc. IEEE PES ISGT*, USA, February 2013.

[13] B. Vaidya and H. Mouftah, "Security mechanism for multi-domain vehicle-to-grid infrastructure," *in Proc. IEEE GLOBECOM*, USA, December 2011.

[14] M. Stegelmann and D. Kesdogan, "V2GPriv: Vehicle-to-grid privacy in the smart grid," *in Proc. CSS*, Australia, December 2012.

[15] H. Guo, Y. Wu and M. Ma, "A batch authentication protocol for V2G communications," *in Proc. NTMS*, France, February 2011.

[16] H. Guo, Y. Wu and M. Ma, "UBAPV2G: A unique batch authentication protocol for vehicle-to-grid communications," *IEEE Transactions on Smart Grid*, vol. 2, no. 4, pp. 707 – 714, November 2011.

[17] H. Tseng, "On the security of a unique batch authentication protocol for vehicle-to-grid communications," *in Proc. ITST*, Taiwan, November 2012.

[18] H. Liu, H. Ning and M. Guizani, "Battery status-aware authentication scheme for V2G networks in smart grid," *IEEE Transactions on Smart Grid*, vol. 4, no. 1, pp. 99 – 110, March 2013.

[19] H. Liu, H. Ning and L. Yang, "Role-dependent privacy preservation for secure V2G networks in the smart grid," *IEEE Transaction on Information Forensics Security*, vol. 9, no. 2, pp. 208–220, February 2014.

[20] Z. Yang, S. Yu and C. Liu, "$P^2$: Privacy-preserving communication and precise reward architecture for V2G networks in smart grid," *IEEE Transactions on Smart Grid*, vol. 2, no. 4, pp. 697 – 706, December 2011.

[21] H. Tseng, "A secure and privacy-preserving communication protocol for V2G networks," *in Proc. IEEE WCNC*, France, April 2012.

[22] D. Ghosh and S. Wicker, "A privacy-aware design for the vehicle-to-grid framework," *in Proc. HICSS*, USA, January 2013.

[23] M. Stegelmann and D. Kesdogan, "Design and evaluation of a privacy-preserving architecture for vehicle-to-grid interaction," *LNCS, Springer Berlin Heidelberg*, vol. 7163, pp. 75 – 90, 2012.

[24] T. Holden and J. Yazdani, "Hybrid security for hybrid vehicles exploring smart grid technology, power-line and wireless communication," *in Proc. IEEE PES ISGT Europe*, UK, December 2011.

[25] H. Su, M. Qiu, and H. Wang, "Secure wireless communication system for smart grid with rechargeable electric vehicle," *IEEE Communications Magazine*, vol. 50, no. 8, pp. 62 – 68, August 2012.

[26] Y. Li, R. Wang and Z. Han, "Resilient PHEV charging policies under price information attacks," *in Proc. IEEE SmartGridComm*, Taiwan, November 2012.

[27] M. Au, J. Liu and J. Zhou, "A new payment system for enhancing location privacy of electric vehicles," *IEEE Transactions on Vehicular Technology*, vol. 63, no. 1, pp. 3 – 18, January 2014.

[28] M. Saarinen, "The BlueJay ultra-lightweight hybrid cryptosystem," *in Proc. IEEE Symposium on Security and Privacy Workshops*, USA, May 2012.

[29] M. Saarinen, "The PASSERINE public key encryption and authentication mechanism," *LNCS, Springer*, vol. 7127, pp. 283 – 288, 2011.

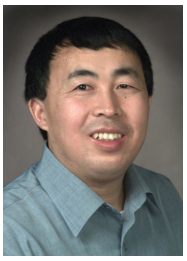
[30] D. Engels, M. Saarinen and E. Smith, "The Hummingbird-2 lightweight authenticated encryption algorithm," *LNCS, Springer*, vol. 7055, pp. 19 – 31, 2011.

[31] A. Menezes and S. Wanstone, *Handbook of Applied Cryptography*. CRC Press, 1996.

[32] H. Martin, E. Millan and J. Castro, "AKARI-x: a pseudorandom number generator for secure lightweight systems," *In Proc. IEEE IOLTS*, Greece, July 2011.

[33] B. Collard and F. Standaert, "A statistical saturation attack against the block cipher PRESENT," *In Proc. CT-RSA*, USA, April 2009. *LNCS, Springer*, vol. 5473, pp. 195 – 210, 2009.


**Asmaa Abdallah** (a3abdall@uwaterloo.ca) received the B.Sc degree in Computer and Control Engineering and M.Sc. degrees in Mobile Networks from Suez Canal University, Egypt in 2003 and 2007, respectively. She is currently a Ph.D. candidate in the Department of Electrical and Computer Engineering, University of Waterloo. Her research interests include Security and Privacy in Smart Grid, Wireless Network Security, and Mobile Computing.



**Xuemin (Sherman) Shen** (IEEE M'97-SM'02-F09) received the B.Sc.(1982) degree from Dalian Maritime University (China) and the M.Sc. (1987) and Ph.D. degrees (1990) from Rutgers University, New Jersey (USA), all in electrical engineering. He is a Professor and University Research Chair, Department of Electrical and Computer Engineering, University of Waterloo, Canada. He is also the Associate Chair for Graduate Studies. Dr. Shen's research focuses on resource management in interconnected wireless/wired networks, wireless network security, social networks, smart grid, and vehicular ad hoc and sensor networks. He is an elected member of IEEE ComSoc Board of Governor, and the Chair of Distinguished Lecturers Selection Committee. Dr. Shen served as the Technical Program Committee Chair/Co-Chair for IEEE Globecom'16, Infocom'14, IEEE VTC'10 Fall, and Globecom'07, the Symposia Chair for IEEE ICC'10, the Tutorial Chair for IEEE VTC'11 Spring and IEEE ICC?08, the General Co-Chair for ACM Mobihoc'15, Chinacom'07 and QShine'06, the Chair for IEEE Communications Society Technical Committee on Wireless Communications, and P2P Communications and Networking. He also serves/served as the Editor-in-Chief for IEEE Network, Peer-to-Peer Networking and Application, and IET Communications; a Founding Area Editor for IEEE Transactions on Wireless Communications; an Associate Editor for IEEE Transactions on Vehicular Technology, Computer Networks, and ACM/Wireless Networks, etc.; and the Guest Editor for IEEE JSAC, IEEE Wireless Communications, IEEE Communications Magazine, and ACM Mobile Networks and Applications, etc. Dr. Shen received the Excellent Graduate Supervision Award in 2006, and the Outstanding Performance Award in 2004, 2007, 2010, and 2014 from the University of Waterloo, the Premier's Research Excellence Award (PREA) in 2003 from the Province of Ontario, Canada, and the Distinguished Performance Award in 2002 and 2007 from the Faculty of Engineering, University of Waterloo. Dr. Shen is a registered Professional Engineer of Ontario, Canada, an IEEE Fellow, an Engineering Institute of Canada Fellow, a Canadian Academy of Engineering Fellow, a Royal Society of Canada Fellow, and a Distinguished Lecturer of IEEE Vehicular Technology Society and Communications Society.