BBARS: Blockchain-Based Anonymous Rewarding Scheme for V2G Networks

Huaqun Wang, Qihua Wang, Debiao He, Qi Li, and Zhe Liu

***Abstract*—In vehicle-to-grid (V2G) networks, battery-powered vehicle (BV) provides service to the power grid. In order to encourage more BVs to provide the service for power grid, it is necessary to reward the BVs from the power grid. To extensively deploy V2G networks, some security and privacy problems must be solved. In this paper, for the first time, we propose the novel concept of blockchain-based anonymous rewarding scheme (BBARS) for V2G networks. The novel concept comes from the application requirement which has not been solved by now. We give the formal system model and security model of BBARS. Then, we design the concrete BBARS scheme by making use of two different public key cryptosystem. Through security analysis and performance analysis, the designed scheme is provably secure and efficient. The analysis results also show the designed BBARS scheme is practical for secure V2G networks in smart grid.**

***Index Terms*—Anonymity, blockchain, reward, smart grid, vehicle-to-grid (V2G) networks.**

Manuscript received August 7, 2018; revised October 4, 2018 and November 19, 2018; accepted December 22, 2018. Date of publication January 1, 2019; date of current version May 8, 2019. The work of H. Wang was supported in part by the National Natural Science Foundation of China under Grant 61872192, in part by the Natural Science Foundation of Jiangsu Province under Grant BK20181394, in part by the Qing Lan Project of Jiangsu Province, in part by the 1311 Talent Plan Foundation of NUPT, and in part by the Opening Project of Shanghai Key Laboratory of Integrated Administration Technologies for Information Security under Grant AGK2018004. The work of D. He was supported by the National Natural Science Foundation of China under Grant 61501333 and Grant 61572379. The work of Z. Liu was supported in part by the National Natural Science Foundation of China under Grant 61802180, in part by the Natural Science Foundation of Jiangsu Province of China under Grant BK20180421, in part by the National Cryptography Development Fund under Grant MMJJ20180105, and in part by the Fundamental Research Funds for the Central Universities under Grant NE20181 06. *(Corresponding author: Qihua Wang.)*

H. Wang is with the Jiangsu Key Laboratory of Big Data Security and Intelligent Processing, School of Computer Science, Nanjing University of Posts and Telecommunications, Nanjing 210023, China, and also with the Shanghai Key Laboratory of Integrated Administration Technologies for Information Security, Shanghai Jiao Tong University, Shanghai 200240, China (e-mail: wanghuaqun@aliyun.com).

Q. Wang is with the School of Medical Information Engineering, Jining Medical University, Rizhao 272067, China (e-mail: wd19791209@163.com).

D. He is with the School of Cyber Science and Engineering, Wuhan University, Wuhan 430072, China (e-mail: hedebiao@163.com).

Q. Li is with the Jiangsu Key Laboratory of Big Data Security and Intelligent Processing, School of Computer Science, Nanjing University of Posts and Telecommunications, Nanjing 210023, China (e-mail: liqics@njupt.edu.cn).

Z. Liu is with the College of Computer Science and Technology, Nanjing University of Aeronautics and Astronautics, Nanjing 211106, China (e-mail: sduliuzhe@gmail.com).

Digital Object Identifier 10.1109/JIOT.2018.2890213

I. Introduction

ALONG with the fast development of information technology and power grid, smart grid becomes a reality. Smart grid consists of all kinds of operational and energy measures, for example, smart meters, smart appliances, renewable energy resources, energy efficient resources, etc. As an important component part, vehicle-to-grid (V2G) networks describe a system in which plug-in electric vehicles, such as battery-powered vehicle (BV), communicate with the power grid to sell demand response services by either returning electricity to the grid or by throttling their charging rate. V2G networks develop very fast in the recent years. Future battery developments may change the economic equation, making it advantageous to use newer high capacity and longer-lived batteries in BV. These newer batteries can be used in grid load balancing and as a large energy cache for renewable grid resources. In May 2016, Nissan and Enel power company announced a collaborative V2G trial project in the United Kingdom, the first of its kind in the country [1]. The trial comprises 100 V2G charging units to be used by Nissan Leaf and e-NV200 electric van users. The project claims electric vehicle owners will be able to sell stored energy back to the grid at a profit. Most vehicles are parked 95% of the time on average, their batteries could be used to let electricity flow from the car to the power lines and back, with a value to the utilities of up to $4000 per year per BV [2]. But, security and privacy preservation are always the obstacles to deploy V2G networks widely, especially the anonymity for BV.

Recently, blockchain develops very fast. It has the following properties: no central control, distributed ledger, and security protection by making use of the cryptography [3]. Blockchain is the core technique of cryptocurrency. It is a growing list of records, called blocks, which are linked using cryptography. Each block contains a cryptographic hash of the previous block, a timestamp, and transaction data (generally represented as a merkle tree root hash). The basic structure of blockchain is shown in Fig. 1. The blockchain takes use of two cryptographic tools: 1) digital signature and 2) cryptographic hash function. A digital signature can authenticate a digital message. A cryptographic hash function has the following properties: preimage resistance, second preimage resistance, and collision resistance. In the blockchain for Bitcoin, a SHA-256 hash function is used. The two words "preimage" and "image" are defined below.

$f : X \to Y$ is a function from the set X to the set Y. The preimage of a set $B \subseteq Y$ under f is the subset of X defined by $f^{-1}[B] = \{x \in X | f(x) \in B\}$. The image of a subset $A \subseteq X$

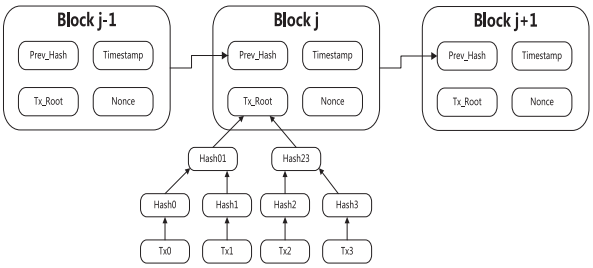


Fig. 1. Basic structure of blockchain.

under f is the subset $f[A] \subseteq Y$ defined by $f[A] = \{y \in Y | y = f(x)$ for some $x \in A\}$.

By making use of proof-of-work (PoW), proof-of-stake, etc., the transaction records are written on the blockchain. PoW makes use of the preimage resistance of hash function to realize the consensus. PoW involves scanning for a value that when hashed, such as with SHA-256, the hash begins with a number of zero bits. The average work required is exponential in the number of zero bits required and can be verified by executing a single hash [3]. The scanned value is the preimage of the hash with a number of zero bits. The hash value is also called as the image. Based on the blockchain, many kinds of digital currencies are designed. For the first digital currency, i.e., Bitcoin, it uses public key cryptography technique to store assets and perform the transactions. By making use of the zero-knowledge proof system, Zerocash was proposed [4], [5]. Wang *et al.* [6] studied the designated-verifier proof of assets for bitcoin exchange. Blockchain can also be used to protect network security and privacy, to make transaction more efficient, to realize digital forensics, etc.

It is an interesting topic to realize the security and privacy problems for V2G networks by making use of blockchain. For the rewarding scheme of V2G, the application requires us to solve the BV anonymity, central aggregator (CAG) anonymity, untraceability, and unlinkability simultaneously. This paper solves this open problem by making use of the novel technique-blockchain.

A. Motivation

Security and privacy problems prevent V2G networks from developing in the smart grid. In order to accept the service of the BVs, it is important to monitor each BV's current status which includes location, battery state of charge, battery capacity, etc. Usually, the task is performed by the V2G network operators. After that, BVs provide ancillary services to the power grid. In order encourage more BVs to provide the services, it is necessary to reward the valid BVs. At the same time, privacy-preservation is also an important concern for the BVs which have the ability to provide the service for the power grid. If the BV' identity is faced with leakage when it gets the rewarding, its enthusiasm will reduce rapidly. Thus, anonymous rewarding is important for the rapid development of V2G networks.

In the real world, there exist some dishonest BVs. They wish to be rewarded more times for one time service. It will incur some disputes between the BVs and the power grid. These disputes maybe come from the opaque management of power grid. Sometimes, BVs have no confidence in the power grid. These disputes crack down on the enthusiasm of BVs and the power grid. The trust and transparent management can advance the cooperation between BVs and the power grid. Anonymity, transparent management and BVs' dishonest demanding are paradoxical requirements which have not been solved before the emerging of blockchain. Although there exist some research results in V2G networks, they do not solve these three paradoxical requirements simultaneously. Furthermore, it is also important to preserve the privacy of power grid. In the existing research results, power grid's bank account is faced with the risk of leaking to BVs. The power grid is also reluctant to its bank account leakage.

In order preserve the privacy of the power grid and BVs and guard against the disputes between the BVs and the power grid, we propose the blockchain-based anonymous rewarding scheme (BBARS) for V2G networks.

B. Related Works

Along with the development of smart grid, the research in V2G networks becomes the novel hot point. In the V2G networks, the incentives mechanism has attracted many researchers' interests. In 2011, Yang *et al.* [7] proposed a very interesting privacy-preserving communication and precise reward architecture for V2G networks. They proposed a type of precise and equitable incentive model with better usability, where the operator of a V2G network rewards each participating BV for each service it provides. They made the first attempt to address privacy considering the very specific nature of V2G networks. In 2015, Wang *et al.* [8] found Yang *et al.*'s scheme is insecure and proposed the concrete attack method. Then, they proposed a new traceable privacy-preserving communication and precise reward scheme with available cryptographic primitives. Regretfully, Wang *et al.*'s scheme cannot satisfy the unlinkability. Due to BV's mobility, the privacy preservation problem is more impressible in V2G networks. Han *et al.* studied various privacy preservation problems in V2G networks, including anonymous authentication, location privacy, identification privacy, concealed data aggregation, privacy-preserving billing and payment, and privacy-preserving data publication. These techniques include homomorphic encryption, blind signature, group signature, ring signature, third party anonymity, and anonymity networks [9]. Their paper does not design new scheme and the surveyed schemes cannot be used in the anonymous rewarding for V2G networks. Wan *et al.* [10] proposed a privacy-preserving solution for V2G communications known as privacy via randomized anonymous credentials. Regretfully, Wan *et al.*'s scheme cannot satisfy unconditional anonymity and unlinkability. Although there are a lot of research results in this field, it is still an open problem to solve the anonymity, unlinkability, and untraceability simultaneously. In order to solve the security problems, many fundamental theories have been studied in this field [11]–[14].

Blockchain was invented by Satoshi Nakamoto in 2008 for use in the cryptocurrency bitcoin, as its public transaction

ledger [3]. It is different from the other network techniques, such as neural network [15], [16], gene network [17], etc. The invention of the blockchain for bitcoin made it the first digital currency to solve the double-spending problem without the need of a trusted authority or central server. Gao *et al.* proposed a blockchain-based privacy preserving payment mechanism for V2G networks, which enables data sharing while securing sensitive user information. The mechanism introduces a registration and data maintenance process that is based on a blockchain technique, which ensures the anonymity of user payment data while enabling payment auditing by privileged users [18]. Dorri *et al.* [19] proposed a blockchain-based architecture to protect the privacy of users and to increase the security of the vehicular ecosystem. Although they consider the privacy of users and the security of the vehicular ecosystem, they do not give the concrete and practical scheme. Through proposing an effective announcement network called CreditCoin, Li *et al.* [20] proposed a novel privacy-preserving incentive announcement network based on blockchain via an efficient anonymous vehicular announcement aggregation protocol. By making use of meters as nodes in a distributed network which encapsulates meter measurements as blocks, Liang *et al.* [21] presented a comprehensive discussion on how blockchain technology can be used to enhance the robustness and security of the power grid. Li *et al.* proposed a secure energy trading system named energy blockchain. This energy blockchain can be widely used in general scenarios of peer-to-peer energy trading getting rid of a trusted intermediary. They also proposed a credit-based payment scheme to support fast and frequent energy trading [22]. Aitzhan and Svetinovic studied transaction security in decentralized smart grid energy trading without reliance on trusted third parties. They have also implemented a proof-of-concept for decentralized energy trading system using blockchain technology, multisignatures, and anonymous encrypted messaging streams, enabling peers to anonymously negotiate energy prices and securely perform trading transactions [23].

From the existing research results, it is clear that it is a trend to solve the security and privacy problem for V2G in smart grid.

C. Our Contributions

Based on the application requirement, this paper studies the vital security and privacy issues of V2G networks in smart grid. Our contributions are listed below.

1) For the first time, we propose the concept of anonymous rewarding scheme for V2G networks by making use of blockchain. We give the formal system model and security model. For the first time, we study the CAG anonymity and BV anonymity simultaneously in the rewarding scheme for V2G networks.
2) The unlinkability is satisfied between the payer address and payee address. The untraceability is also satisfied. No one can establish their relation between them when the reward is performed. No one can find the payer address.
3) Two different PKCs are used in the design of our concrete BBARS scheme. In order to improve the efficiency, we take use of the aggregate signature by making use of the PKCs in public key infrastructure (PKI). At the same time, another different PKCs is also used when the operations are performed on the blockchain.
4) We propose the first BBARS scheme with available cryptographic primitives. The proposed BBARS scheme is provably secure in the random oracle model. Theoretical analysis and experimental analysis demonstrate that our scheme is efficient and practical for secure V2G networks in smart grid.

D. Organization

The organization of this paper is listed below. Section II gives some cryptographic background which includes PKCs in PKI, PKCs on the blockchain, elliptic curve public key cryptography (ECC), bilinear pairings, aggregate signature, and ring signature. Section III introduces the system model and security model of our BBARS concept. Section IV designs an efficient BBARS scheme. Section V gives a detailed security and performance analysis of our BBARS scheme. Finally, Section VI concludes this paper.

II. Background

In this section, we will review PKCs in PKI, PKCs on the blockchain, ECC, bilinear pairings, aggregate signature, and ring signature. In this paper, the cardinality of a set S is defined as $|S|$.

A. PKCs in PKI, and PKCs on the Blockchain

PKI is a set of roles, policies, and procedures needed to create, manage, distribute, use, store, and revoke digital certificates and manage public-key encryption. PKI can facilitate the secure electronic transfer of information for a lot of network transactions, such as confidential email, e-commerce, electronic banking, etc. PKCs is a cryptographic system which has a pair of keys: public key which may be disseminated widely, and secret key which is known only to the owner. In PKI, PKI binds public key with respective identities of entities. The binding is established through a process of registration and issuance of certificates by a certificate authority (CA).

On the blockchain, there are no trusted third parties. Concretely, there are no PKI, key generation center, CA, etc. For the PKCs on the blockchain, the user picks the secret key/public key pair. The public key corresponds to the address on the blockchain. His assets are stored on the address. Because the user has the corresponding secret key, he can make use of the assets on the address. According to the basic hypotheses of microeconomics, i.e., hypothesis of rational man, the user does not discard the assets on the address.

B. ECC and Bilinear Pairings

For ECC, the operations are performed on the elliptic curve group E over the finite field $\mathbb{F}_q$ where q is a prime number.

$E(\mathbb{F}_q)$ can be given by the following equation: $E: y^2 = x^3 + ax + b \bmod q$ where $a, b \in \mathbb{F}_q^*$ and $4a^3 + 27b^2 \bmod q \neq 0$. Let G be a generator of $E(\mathbb{F}_q)$ with the prime order $\hat{l}$. On the elliptic cure $E(\mathbb{F}_q)$, the discrete logarithm problem (DLP) and computational Diffie–Hellman problem (CDHP) are given below.

Definition 1 (DLP): Given the pair $(G, G_1) \in E(\mathbb{F}_q)$, DLP is to compute $x \in \mathbb{F}_{\hat{l}}^*$ such that $G_1 = xG$.

Definition 2 (CDHP): Given the triple $(G, G_1, G_2) \in E(\mathbb{F}_q)^3$, CDHP is to compute xyG where $G_1 = xG$, $G_2 = yG$ and x, y are unknown.

A well-known theorem of Hasse states that $|E(\mathbb{F}_q)| = q + 1 - t$, where $|t| \leq 2^{0.5}$. The curve E is said to be supersingular if $t^2 = 0, q, 2q, 3q, 4q$; otherwise, the curve is non-supersingular. For ECC in this paper, we choose non-supersingular elliptic curve E. On the corresponding elliptic curve group $E(\mathbb{F}_q)$, DLP, CDHP are assumed to be computationally difficult [24], [25].

In this paper, we also take use of the bilinear group pairs $(\mathbb{G}_1, \mathbb{G}_2)$. Let $\mathbb{G}_1$ and $\mathbb{G}_2$, respectively, be a cyclic additive group generated by P and a cyclic multiplicative group with the same prime order p. Let $e: \mathbb{G}_1 \times \mathbb{G}_1 \to \mathbb{G}_2$ be a bilinear map with the following properties.

1) *Bilinearity:* $\forall Q, R, S \in \mathbb{G}_1$ and $a, b \in \mathbb{F}_p^*$

$$e(R, Q + S) = e(Q + S, R) = e(Q, R)e(S, R),$$
$$e(aR, bQ) = e(R, Q)^{ab}.$$

2) *Nondegeneracy:* $\exists Q, R \in \mathbb{G}_1$ such that $e(Q, R) \neq 1_{\mathbb{G}_2}$.
3) *Computability:* $\forall Q, R \in \mathbb{G}_1$, there is an efficient algorithm to compute $e(Q, R)$.

Such a bilinear map e can be constructed using the modified Weil [26] or Tate pairings [27] on supersingular elliptic curve $\mathbb{G}_1$. A group $\mathbb{G}_1$ with such a map e is called a bilinear group, on which the CDHP is assumed hard while the decisional Diffie–Hellman problem (DDHP) is easy [28]. Namely, given unknown $a, b, c \in \mathbb{F}_p^*$ and $P, aP, bP, cP \in \mathbb{G}_1$, it is known that there exists an efficient algorithm to determine whether $ab = c \bmod p$ by verifying $e(aP, bP) \stackrel{?}{=} e(P, cP)$ in polynomial time (DDHP), while there exist no efficient algorithms to compute $abP \in \mathbb{G}_1$ with non-negligible probability within polynomial time (CDHP).

C. Aggregate Signature, Ring Signature and Monero

The concept of aggregate signature was introduced by Boneh *et al.* [29] in Eurocrypt 2003. An aggregate signature scheme is a signature scheme which can aggregate many signatures on many distinct messages from many distinct users into one single signature. The validity of an aggregate signature can convince the verifier that these users indeed sign these original messages. It can reduce the computational cost for the verification process because it aggregates many different signatures into one single aggregated signature. Due to the function, aggregate signature can be used in many fields [30], [33]. It can be used in our BBARS scheme.

The concept of ring signature was introduced in 2001 by Rivest *et al.* [31]. For a secure ring signature, it is required that only users in the group member list can generate a valid signature and the signatures generated by different members are theoretically indistinguishable. The former property is referred to as unforgeability and the latter as unconditional anonymity. It has been shown that ring signature is a very useful cryptographic primitive in many applications [32]–[34].

By making use of ring signature and random address, Monero was designed [32]. In 2017, Kumar *et al.* [35] studied the traceability of Monero. They found some transactions are performed even if the cardinality n of address ring is 1, i.e., $n = 1$. The untraceability of Monero cannot be ensured. In order to solve this problem, developers of Monero modifies the system and requires $n > 1$ for each transaction. Monero is a cryptocurrency for a connected world. It is fast, private, and secure. Monero transactions are confirmed by distributed consensus and then immutably recorded on the blockchain. Monero uses ring signatures, ring confidential transactions, and stealth addresses to obfuscate the origins, amounts, and destinations of all transactions. It provides all the benefits of a decentralized cryptocurrency, without any of the typical privacy concessions. Sending and receiving addresses as well as transacted amounts are obfuscated by default. Transactions on the Monero blockchain cannot be linked to a particular user or real-world identity. Monero is fungible because it is private by default. Units of Monero cannot be blacklisted by vendors or exchanges due to their association in previous transactions.

III. System Model and Security Model

In this section, we present the system model and security model for our BBARS scheme. It contains the system entities, their roles, security requirements, and formal security model.

A. System Model

In the BBARS for V2G networks, there are four different entities, i.e., CAG, local aggregator (LAG), BV, and the blockchain. Their functions are given below.

1) *CAG:* It is an operator of the V2G networks. It communicates with the electricity market on behalf of the geographically dispersed BVs. At the same time, it also authorizes the legal BVs to provide the services to itself. When receiving the BVs' services, CAG rewards these BVs. In order to guard against the communication bottleneck and computation bottleneck, some LAGs are added to V2G networks. In this case, CAG also communicates with LAG.
2) *LAG:* It is also an operator of the V2G networks. LAG is different from CAG because it is deployed for every local area. LAG directly receives BVs' service within its local area. Then, it sends the receipt to the BV and certificate to CAG, respectively. Receipt and certificate can be used to confirm the BVs' service.
3) *BV:* For each BV parking position, either it belongs to a commercial parking lot or a private residence, a charging device should be deployed, through which the parked BV could connect to the power grid. BV can communicate with the power grid by deploying a charging and discharging equipment.

4) *Blockchain:* CAG sends the rewards to BVs through the blockchain. BVs can also check whether their rewards have been sent by CAG.

B. Security Model

In this paper, we consider more comprehensive privacy issues. The private information comprises BVs' identities, payee address (for BVs) on the blockchain and payer address (for CAG) on the blockchain. Besides of these privacy, our BBARS scheme also satisfies the unlinkability between payee address (for BVs) and payer address (for CAG). In order to preserve the privacy of BVs and CAG, our BBARS scheme must satisfy the following security goals.

1) Mutual authentication among BV, CAG, and LAG. When BV wants to provide the service to the power grid, it must get the authorization from CAG. By checking the validity of the authorization from CAG, LAG decides whether it accepts BV's service. CAG and LAG interacts to get the detailed information of BV's service.
2) Anonymity for BV. In the process of receiving BV's service, LAG cannot identify the BV's identity. In the process of authorization and rewarding, CAG cannot identify the BV's identity.
3) Anonymity for CAG. Although BV accepts CAG's rewards, BV cannot identify CAG's address on the blockchain.
4) Unlinkability between payee address (i.e., BV) and payer address (i.e., CAG).

According to the above security requirements, we give the formal security definitions below.

Definition 3 (Unforgeability): CAG's authentication protocol satisfies the unforgeability if the probability that $\mathcal{A}$ wins the following game is negligible where $\mathcal{A}$ is probabilistic polynomial time adversary.

1) *Setup:* The system parameters are created and CAG's private/public key pair are generated. At the same time, BA's private/public key pair are also generated. For CAG, its private/public key pair are generated in PKI. For BV, its private/public key pair are generated by the BV itself where there are no the trusted third party. System parameters, CAG's public key and BV's public key are sent to $\mathcal{A}$. We denote the public parameters as params.
2) Interaction between $\mathcal{A}$ and the challenger $\mathcal{C}$. In the interaction, $\mathcal{A}$ adaptively queries $\mathcal{C}$ and gets $\mathcal{C}$'s responses. The queries and responses are listed below.
   a) *Hash Query:* $\mathcal{A}$ sends the hash queries to $\mathcal{C}$. $\mathcal{C}$ creates the hash function value and sends it to $\mathcal{A}$ (random oracle model). Or, $\mathcal{C}$ accesses the hash function and responds $\mathcal{A}$ with the real hash value (standard model).
   b) *Authentication Query:* $\mathcal{A}$ makes the authentication query on the different public key with the corresponding message, which are denoted as $\text{Cont}_{\hat{i}}$. $\mathcal{C}$ creates the authentication $\sigma_{\hat{i}}$ and sends it to $\mathcal{A}$. In the process, we denote the query set as $\{\text{Cont}_{\hat{i}}|\hat{i} \in \hat{\mathbb{I}}\}$ and the response set as $\{\sigma_{\hat{i}}|\hat{i} \in \hat{\mathbb{I}}\}$.
3) *Forgery:* $\mathcal{A}$ can forge a valid authentication on a new public key with the corresponding message, which is denote as Cont. The forged public key and metadata are different from the queried public keys with the corresponding messages, i.e., $\text{Cont} \notin \{\text{Cont}_{\hat{i}}|\hat{i} \in \hat{\mathbb{I}}\}$.

We say that $\mathcal{A}$ wins the above game between $\mathcal{A}$ and $\mathcal{C}$ if

$$\Pr\left[\begin{array}{c}\text{Verify (Cont, params)}\\ =\text{“success”}\end{array}\middle|\begin{array}{c}\text{Cont} \notin\\ \left\{\text{Cont}_{\hat{i}}|\hat{i} \in \hat{\mathbb{I}}\right\}\end{array}\right] \geq \frac{1}{p(k)}$$

where $p(k)$ is a polynomial of the security parameter k. In other words, we say that $\mathcal{A}$ wins if $\mathcal{A}$'s success probability is non-negligible.

Notes: The above definition gives the formal definition of unforgeability for CAG authentication. For LAG authentication and ring signature for transaction, the formal definitions of their unforgeability are similar to the above definition. Due to the page limits, we omit the corresponding definitions.

Definition 4 (Anonymity for BV): In our BBARS scheme, BV is unconditionally anonymous. In other words, the adversary cannot identify the BV's real identity even if the adversary's computing power is infinite.

Definition 5 (Anonymity for CAG): Suppose that CAG has only one address when it sends the rewards to BVs. We say our BBARS scheme satisfies CAG anonymity if the following two conditions hold.

1) When CAG has n_1 output addresses, if all the output addresses do not belong to the adversary, the probability that CAG's change address can be identified is not more than $(1/n_1)$. If n_2 output addresses belong to the adversary, the probability that CAG's change address can be identified is not more than $[1/(n_1 - n_2)]$.
2) If CAG computes a transaction on behalf of a ring of n addresses, any adversary not belonging to the ring should not have probability greater than $(1/n)$ to guess the CAG's address. If the adversary has n' addresses of the ring, but not CAG, then his probability of guessing the CAG's address is not greater than $(1/(n - n'))$.

Definition 6 (Untraceability): For each incoming transaction, all possible senders are equiprobable.

Definition 7 (Unlinkability): For any two outgoing transactions, it is impossible to prove they were sent to the same person.

Notes: From the definition of anonymity for CAG, we know that it implies the definition of untraceability. Thus, when we analyze our BBARS scheme's security, the property of traceability is omitted.

We review the basic hypotheses of microeconomics: hypothesis of rational man. The hypothesis of rational man is the basic assumption in traditional economic theories. The hypothesis means that any man is a rational man who is selfish and desires to maximize his needs or desires.

This paper uses many notations. We give these notations and their descriptions in Table I.

IV. Efficient BBARS Scheme

Following the system model and the formal security models, we propose an efficient BBARS for V2G networks. The

TABLE I
NOTATIONS AND DESCRIPTIONS

Notations	Descriptions
E	Elliptic curve over $\mathbb{F}_q$ for Monero
$\mathbb{F}_q, \mathbb{F}_p$	The finite field
$\mathbb{F}_q^*, \mathbb{F}_p^*$	The corresponding multiplicative groups of $\mathbb{F}_q, \mathbb{F}_p$
G	Base point of E
$\hat{l}$	Prime order of G
H_1, H_2	Cryptographic hash functions
(a_i, b_i)	BV$_i$'s random private key
(A_i, B_i)	BV$_i$'s random public key
(x, y)	CAG's private key on the blockchain
(X, Y)	CAG's public key which is also CAG's address on the blockchain
$(\mathbb{G}_1, \mathbb{G}_2)$	Bilinear group pair
p	The prime order of $\mathbb{G}_1$ and $\mathbb{G}_2$
e	Bilinear pairing
P	A generator of $\mathbb{G}_1$
(z, Z)	CAG's private key/public key pair over $\mathbb{G}_1$ in PKI
(l_i, L_i)	LAG$_i$'s private key/public key pair over $\mathbb{G}_1$ in PKI
H	Full-domain cryptographic hash function
$\mathbb{I}$	The set of BV's index
$\mathbb{J}$	The set of LAG's index
$m_{i,j}$	The message that corresponds to BV$_i$'s service through LAG$_j$
$(Cont_i, \sigma_i)$	Authorization for BV$_i$ from CAG
$(m_{i,j}, A_i, B_i, \bar{\sigma}_{i,j})$	The receipt for BV$_i$ from LAG$_j$ and the certificate for CAG from LAG$_j$
$\hat{P}_i$	One-time public key for BV$_i$, which is also BV$_i$'s address on the blockchain
bal_i	The whole rewarding balance for BV$_i$
bal_c	The remaining change for CAG
P_i	The public key of the user U_i, which is also U_i's address on the blockchain
$\|S\|$	The cardinality of the set S

proposed scheme takes use of some cryptographic techniques which include digital signature, ring signature, encryption, blockchain, and Monero.

Our proposed scheme must satisfy the security requirements of anonymity and transparent management. At the same time, it can also avoid the dishonest BVs' illegal demanding. In order to get a secure and efficient scheme, some complex cryptographic techniques are used. The proposed scheme consists of seven procedures: 1) setup; 2) *contract*-based authorization; 3) anonymous service provision and reward; 4) reward from CAG; 5) verification and gain; 6) solve the dispute; and 7) BV revocation. The detailed procedures are given next.

In order to show the intuition of the scheme's design, the architecture of our concrete scheme is presented in Fig. 2. There are four entities, i.e., Blockchain, CAG, LAG, and BV, which interact among them (circled numbers in the sequel correspond to numbers in Fig. 2. We assume that the system has been setup and the system parameters have been generated. Based on the preconditions, we give the outline of their interactions. ① BVs and CAG generate the private/public key pair, respectively. The public keys correspond to the addresses on the blockchain. At the same time, CAG and LAG also generate the other private/public key pairs in PKI. ② By making use of BV's public key, BV registers it at CAG. ③ CAG authorizes LAG to accept some service from the valid BVs. At the same time, it also informs LAG the invalid BVs. That is to say, CAG has the right to revoke some BVs. ④ BV provides service for the power grid through LAG. LAG sends the receipt to BV. At the same time, LAG generates the certificate for the service of BV. ⑤ When the time slot ends, LAG summarized these certificates and sends them to CAG. ⑥ CAG verified LAG's certificates. When they are valid, CAG sends the rewards to BV through the blockchain.

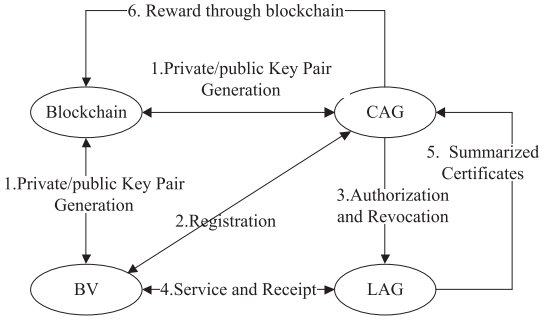


Fig. 2. Architecture of our rewarding scheme.

A. Setup

In our system, we use two types of system parameters. One type parameters are for the blockchain and Monero. The other type parameters are for the registration, receipt, certificate, authorization, and revocation. They are generated below.

1) We take use of the parameters which are the same as Monero. The elliptic curve $E: -x^2 + y^2 = 1 + dx^2y^2$ is defined on the finite field $\mathbb{F}_q$, where $q = 2^{255} - 19$, $d = -122665/121666$. Pick a base point G which has the prime order $\hat{l} = 2^{252} + 27742317777372353535851937790883648493$. Two cryptographic hash functions are given below: $H_1 : \{0,1\}^* \to \mathbb{F}_{\hat{l}}$, $H_2 : E(\mathbb{F}_q) \to E(\mathbb{F}_q)$. BV$_i$ chooses a pair $(a_i, b_i) \in \mathbb{F}_{\hat{l}}^*$ as its private key. The corresponding public key is the two computed points (A_i, B_i) where $A_i = a_iG, B_i = b_iG$. CAG chooses $x \in \mathbb{F}_{\hat{l}}^*$ as its private key. The corresponding public key is the computed points X where $X = xG$. These public keys are transformed into the human friendly string with error correction. The human friendly strings are used as the standard address on the blockchain. In order to save the symbol number, we denote the public key as their address on the blockchain. For the same reason, the digital assets and rewards are also expressed as the Monero. In the phase, CAG stores a lot of Monero on the address X of the blockchain.
2) In PKI, the following parameters are generated. Let $(\mathbb{G}_1, \mathbb{G}_2)$ be a bilinear group pair of prime order p. Let the bilinear map be $e : \mathbb{G}_1 \times \mathbb{G}_1 \to \mathbb{G}_2$. Let P be a generator of the group $\mathbb{G}_1$. CAG chooses its private key $z \in \mathbb{F}_p^*$ and computes its public key $Z = zP$. LAG$_i$ chooses its private key $l_i \in \mathbb{F}_p^*$ and computes its public key $L_i = l_iP$.

The public key and the corresponding entity's identity are linked through the certification which is issued by the PKI. Let H be a full-domain cryptographic hash function $H : \{0, 1\}^* \to \mathbb{G}_1$.

3) Pick a secure signature/verification algorithm pair (Sign, Verify).

B. Contract-Based Authorization

In order to provide the service and get the reward, the ith BV, i.e., BV_i, needs CAG's agreement. At the same time, CAG also needs to collect BV_i's status information. Based on the these information, the agreed contract Cont_i is created. Cont_i contains BV_i's status information, reward for the service, service charge standards and payment method, etc. In order to preserve BV_i's privacy, Cont_i does not contain BV_i' identity information. But, (A_i, B_i) is contained into Cont_i to get the authorization from CAG. A table Tab is initialized which lists the authorized BV_i's random public key and period of service. CAG performs the following procedures to authorize the BV_i.

1) CAG generates the signature σ_i for Cont_i below: $\sigma_i = zH(\mathrm{Cont}_i)$.
2) CAG adds the address (A_i, B_i) and period of service into Tab.
3) CAG sends $(\mathrm{Cont}_i, \sigma_i)$ to BV_i.
4) CAG sends the updated table Tab to all the LAGs.

C. Anonymous Service and Receipt

When BV_i accesses the V2G network and provides the service for the power grid, it presents $(\mathrm{Cont}_i, \sigma_i)$ to the LAG_j in the local area, where $j \in \mathbb{J}$. LAG_j's private/public key pair is (l_j, L_j) where $L_j = l_jP$. $(\mathrm{Cont}_i, \sigma_i)$ can be verified below.

1) At some moment, LAG_j receives a lot of pairs $(\mathrm{Cont}_i, \sigma_i)$ where $i \in \mathbb{I}$.
2) LAG_j picks the random numbers $\alpha_i \in \mathbb{F}_p, i \in \mathbb{I}$ and verifies whether the following formula holds: $e(\sum_{i\in\mathbb{I}} \alpha_i\sigma_i, P) = e(\sum_{i\in\mathbb{I}} \alpha_i H(\mathrm{Cont}_i), Z)$. If the formula does not hold, LAG_j finds out the invalid pairs and reject them, then go to the following procedure; otherwise, LAG_j performs the following procedure.
3) For every $i \in \mathbb{I}$, LAG_j extracts BV_i's public keys (A_i, B_i) from Cont_i. If (A_i, B_i) belongs to Tab and the corresponding period of service is valid, LAG_j accepts BV_i's service; otherwise, LAG_j rejects BV_i's service.
4) When BV_i provides service for the power grid, LAG_j generates receipt below.
   a) Denote BV_i's service and the corresponding reward as the message $m_{i,j}$. LAG_j computes $\bar{\sigma}_{i,j} = l_jH(m_{i,j}, A_i, B_i)$.
   b) LAG_j sends $(m_{i,j}, A_i, B_i, \bar{\sigma}_{i,j})$ to CAG and BV_i, respectively.

Notes: For BV_i, $(m_{i,j}, A_i, B_i, \bar{\sigma}_{i,j})$ is the receipt for its service. For CAG, $(m_{i,j}, A_i, B_i, \bar{\sigma}_{i,j})$ is the certificate to reward BV_i.

D. Reward From CAG

Suppose that there are many LAGs. We denote them as $\{\mathrm{LAG}_j, j \in \mathbb{J}\}$. For LAG_j, the corresponding private/public key pair is (l_j, L_j) where $L_j = l_jP$. In order to reward the BVs for their valid services, CAG performs the procedures below.

1) During some period, CAG receives the certificates $(m_{i,j}, A_i, B_i, \bar{\sigma}_{i,j})$ from LAG_j, where $i \in \mathbb{I}, j \in \mathbb{J}$. CAG picks the random numbers $\beta_{i,j} \in \mathbb{F}_p, i \in \mathbb{I}, j \in \mathbb{J}$ and verifies them by checking whether the following formula holds:
$$e\left(\sum_{j\in\mathbb{J},i\in\mathbb{I}} \beta_{i,j}\bar{\sigma}_{i,j}, P\right) = \prod_{j\in\mathbb{J},i\in\mathbb{I}} e\big(\beta_{i,j}H\big(m_{i,j}, A_i, B_i\big), L_j\big).$$
If the above formula does not hold, CAG finds out the invalid pairs and rejects them; otherwise, CAG goes to the next step.
2) For the BV_i where $i \in \mathbb{I}$, CAG unpacks the received messages and gets its address (A_i, B_i). CAG generates a random number $r \in \mathbb{F}_l^*$, then it gets a one-time public key $\hat{P}_i = H_1(rA_i)G + B_i$ for BV_i.
3) For the messages $m_{i,j}$ where $i \in \mathbb{I}$, $j \in \mathbb{J}$, CAG can unpack it and get the rewarding balance $\mathrm{bal}_{i,j}$. Then, CAG gets the whole rewarding balance $\mathrm{bal}_i = \sum_{j\in\mathbb{J}} \mathrm{bal}_{i,j}$. Suppose that CAG has the balance bal on the blockchain. Thus, it prepares $|\mathbb{J}| + 1$ outputs in one transaction. Concretely, for BV_i whose one-time public key (i.e., address) is $\hat{P}_i$, the output corresponds to the rewarding balance bal_i. Besides them, the additional output corresponds to the change $\mathrm{bal}_c = \mathrm{bal} - \sum_{i\in\mathbb{I}} \mathrm{bal}_i$. In order to simply the symbols, we denote the $|\mathbb{J}| + 1$ outputs and some metadata as the message m. For example, m contains R and $(\hat{P}_i, \mathrm{bal}_i)$ for every $\mathrm{BV}_i, i \in \mathbb{I}$ where $R = rG$.
4) CAG calculates $A = H_2(\mathrm{Sign}_z(m))$ where $\mathrm{Sign}_z(m)$ is the signature on the message m by making use of CAG's private key z.
5) CAG selects a random subset $\mathcal{S}'$ of the other users' public key P_i and $\mathcal{S}'$ has the cardinality n, his own private/public key pair is (x_s, P_s) where $0 \le s \le n$. It also computes the image $I = x_sH_2(P_s)$. It picks the random numbers $\{q_i | i = 0, 1, \ldots, n\}$ and $\{\omega_i | i = 0, 1, \ldots, n, i \ne s\}$ from $\mathbb{F}_l^*$. Then, it computes the following points:
$$L_i = \begin{cases} q_iG, & \text{if } i = s \\ q_iG + \omega_iP_i, & \text{if } i \ne s \end{cases}$$
$$R_i = \begin{cases} q_iH_2(P_i), & \text{if } i = s \\ q_iH_2(P_i) + \omega_iI, & \text{if } i \ne s. \end{cases}$$
Then, CAG computes
$$c = H_1(m, A, L_0, \ldots, L_n, R_0, \ldots, R_n)$$
and the following values:
$$c_i = \begin{cases} \omega_i, & \text{if } \ i \ne s \\ c - \sum\limits_{i\ne s} c_i, & \text{if } \ i = s \end{cases}$$
$$r_i = \begin{cases} q_i, & \text{if } \ i \ne s \\ q_s - c_sx_s, & \text{if } \ i = s. \end{cases}$$
The resulting signature is
$$\sigma = (I, A, c_0, \ldots, c_n, r_0, \ldots, r_n).$$

When CAG finished the above procedures, it sends the coin bal_i to $\hat{P}_i$ and bal_c to $\hat{P}_c$ with the signature σ where $\hat{P}_c$ is selected by CAG. $\hat{P}_c$ is used to store the change bal_c.

Notes: Based on the property of Monero transaction, for each record on the blockchain, the corresponding private key can be used only one time. It can avoid double-spending. Thus, the change bal_c must be moved to new address on the blockchain. The new address $\hat{P}_c$ is chosen by CAG. Since CAG knows $\hat{P}_c$'s private key, CAG still owns the challenge bal_c.

E. Verification and Gain

Upon receiving the signature σ for the message m, the verifier performs the following procedures to check the validity of the signature σ.

1) The verifier computes

$$L_i' = r_i G + c_i P_i, \;\; R_i' = r_i H_2(P_i) + c_i I.$$

2) The verifier checks whether the following formula holds:

$$\sum_{i=0}^{n} c_i = H_1\big(m, A, L_0', \ldots, L_n', R_0', \ldots, R_n'\big) \bmod \hat{l}.$$

If it does not hold, the signature is rejected; otherwise, the verifier goes to the next step.

3) The verifier checks whether I has been used in past signatures. If it appeared in the past signatures, the signature is rejected; otherwise, the verifier accepts σ and finishes all the outputs. In other words, the coin bal_i is moved to $\hat{P}_i$ and bal_c is moved to $\hat{P}_c$ from the address P_s.

BV_i checks CAG's reward (m, σ). From m, BV_i extracts R and $(\hat{P}_i, \text{bal}_i)$ where $i \in \mathbb{I}$. BV_i computes $P_i' = H_1(a_i R)G + B_i$. If $P_i' \in \{\hat{P}_i, i \in \mathbb{I}\}$, there exists $\hat{i} \in \mathbb{I}$ which satisfies $P_i' = P_{\hat{i}}$. In order to gain the reward $\text{bal}_{\hat{i}}$, BV_i computes $x_i = H_1(a_i R) + b_i$ which satisfies $P_{\hat{i}} = x_i G$. Thus, BV_i gains the reward $\text{bal}_{\hat{i}}$. Since BV_i knows the private key of the address $P_{\hat{i}}$, BV_i gains the reward $\text{bal}_{\hat{i}}$.

F. Solve the Dispute

When BV_i cannot find its rewards, it sends its receipt to CAG. CAG checks BV_i's receipt, if it is valid, CAG tells it the transaction information with the ring signature. If BV_i thinks CAG is not the real signer, CAG performs the following procedure to show it is the real signer.

1) CAG shows BV_i the preimage $\text{Sign}_z(m)$ of the hash function H on the image A.
2) BV_i verifies whether $\text{Sign}_z(m)$ is the preimage of A. BV_i verifies whether $\text{Sign}_z(m)$ is a valid signature by making use of the verification algorithm Verify with CAG's public key Z in PKI. If these two verifications are valid, BV_i admits CAG is the real signer; otherwise, BV_i denies CAG is the real signer.

G. BV Revocation

We consider BV revocation schemes from three cases.

- *Case 1:* When LAG wants to reject BV's service, LAG refuses BV's service request. When CAG wants to reject BV's service, CAG updates the table Tab. CAG adds the revocation information to Tab and sends it to LAGs.
- *Case 2:* When some BV would like to be revoked, BV would refuse to provide the service to GAG. It can also informs this information to CAG. CAG updates the table Tab and sends it to LAGs.
- *Case 3:* When BV's service expired, it is natural to successfully realize the BV revocation.

V. Security and Performance Analysis

This section, analyzes our BBARS scheme's security and performance. According to security analysis and performance analysis, our BBARS scheme is secure and efficient. It also implies that our scheme is practical.

A. Security Analysis

Theorem 1 (Correctness): If CAG and LAG are honest and follow the proposed scheme, BV_i's authorization from CAG can pass LAG's verification. At the same time, LAG's certificates can pass CAG's verification.

Proof: According to the generation procedures of BV_i's authorization and LAG's certificates, we get:

1) *correctness for BV_i's authorization:*

$$\begin{aligned} e\left(\sum_{i\in\mathbb{I}} \alpha_i \sigma_i, P\right) &= \prod_{i\in\mathbb{I}} e(\sigma_i, P)^{\alpha_i} \\ &= \prod_{i\in\mathbb{I}} e(zH(\text{Cont}_i), P)^{\alpha_i} \\ &= \prod_{i\in\mathbb{I}} e(\alpha_i H(\text{Cont}_i), zP) \\ &= e\left(\sum_{i\in\mathbb{I}} \alpha_i H(\text{Cont}_i), Z\right) \end{aligned}$$

2) *correctness for LAG's certificates:*

$$\begin{aligned} \left(\sum_{j\in\mathbb{J}, i\in\mathbb{I}} \beta_{i,j} \bar{\sigma}_{i,j}, P\right) &= \prod_{j\in\mathbb{J}, i\in\mathbb{I}} e\big(\beta_{i,j} \bar{\sigma}_{i,j}, P\big) \\ &= \prod_{j\in\mathbb{J}, i\in\mathbb{I}} e\big(\beta_{i,j} l_j H\big(m_{i,j}, A_i, B_i\big), P\big) \\ &= \prod_{j\in\mathbb{J}, i\in\mathbb{I}} e\big(\beta_{i,j} H\big(m_{i,j}, A_i, B_i\big), l_j P\big) \\ &= \prod_{j\in\mathbb{J}, i\in\mathbb{I}} e\big(\beta_{i,j} H\big(m_{i,j}, A_i, B_i\big), L_j\big). \end{aligned}$$

■

Theorem 2 (Correctness): If CAG and the verifier are honest and follow the proposed scheme, CAG's signature can pass the verifier's verification.

Proof: From the generation process of CAG's signature, we know the following.

1) When $i \neq s$, we get

$$L_i' = r_i G + c_i P_i = q_i G + \omega_i P_i = L_i$$

and

$$R_i' = r_i H_2(P_i) + c_i I = q_i H_2(P_i) + \omega_i I = R_i.$$

2) When $i = s$, we get

$$L'_s = r_s G + c_s P_s = (q_s - c_s x_s)G + c_s P_s = q_s G = L_s$$

and

$$\begin{aligned} R'_s &= r_s H_2(P_s) + c_s I \\ &= (q_s - c_s x_s) H_2(P_s) + c_s I \\ &= q_s H_2(P_s) \\ &= R_s. \end{aligned}$$

Based on the above 1) and 2), we get

$$\begin{aligned} \sum_{i=0}^{n} c_i &= c \\ &= H_1(m, A, L_0, L_1, \ldots, L_n, R_0, R_1, \ldots, R_n) \\ &= H_1\left(m, A, L'_0, L'_1, \ldots, L'_n, R'_0, R'_1, \ldots, R'_n\right) \bmod \hat{l}. \end{aligned}$$

■

Theorem 3 (Unforgeability): In our BBARS scheme, BV_i's authorization from CAG, the receipt and certificate from LAG, and the ring signature from CAG satisfy the unforgeability.

Outline of Proof: In order to give the efficient authorization algorithm, receipt and certificate generation algorithms, we take use of Boneh *et al.*'s short signature scheme in PKI. Concretely, Boneh *et al.*'s short signature scheme is provably secure and the detailed proof process has been given in [28]. For the ring signature from CAG, we take use of Saberhagen's transaction scheme [32]. The difference from their scheme is that we add the hash function value A into the final ring signature. The proof process is very similar to Saberhagen's proof process. We omit them due to the page limits. ■

Theorem 4 (Anonymity for BV): In our BBARS scheme, BV is unconditionally anonymous. Even if the adversary's computing power is infinite, it cannot identify BV's real identity.

Proof: In the phase of contract-based authorization, the submitted BV_i's information Cont_i does not include its identity information. The chosen public keys (A_i, B_i) have nothing to do with BV_i's identity. In the phase of anonymous service and receipt, BV_i submits $(\mathrm{Cont}_i, \sigma_i)$ to LAG_j. The submitted message also has nothing to do with BV_i's identity. Thus, our BBARS scheme does not need BV_i's identity. Our scheme satisfies the anonymity for BV_i. ■

Theorem 5 (Anonymity for CAG): Our BBARS scheme satisfies anonymity for CAG.

Proof: Based on the Definition 5, we prove this theorem from the following two parts.

1) When CAG has n_1 output addresses and all the output addresses do not belong to the adversary, all the one-time public key $\hat{P}_i = H_1(rA_i)G + B_i$ is random due to the property of the hash function H_1. BV_i's change is transferred to a randomly chosen address on the blockchain. Thus, all the n_1 output addresses are random for the adversary, the probability that CAG's change address can be identified is not more than $(1/n_1)$. When n_2 output addresses belong to the adversary, the other $n_1 - n_2$ output addresses are still random for the adversary. The probability that CAG's change address can be identified is not more than $(1/(n_1 - n_2))$.
2) Our phase *Reward from CAG* takes use of the ring signature idea to realize the anonymity of CAG. From the final signature $\sigma = (I, c_0, \ldots, c_n, r_0, \ldots, c_n)$, we know that $I = x_s H_2(P_s)$, $c_i, r_i, i \neq s$ are random. On the other hand, both $c_s = H_1(m, A, L_1, \ldots, L_n, R_1, \ldots, R_n) - \sum_{i \neq s} c_i$ and $r_s = q_s - c_s x_s$ are random. Thus, if CAG computes a transaction on behalf of a ring of n addresses, CAG's anonymity can be satisfied. ■

In the following part, we analyze why our BBARS scheme can satisfy the unlinkability and solve the dispute.

1) From the phase *Reward from CAG*, the outgoing transaction addresses are one-time public key $\hat{P}_i = H_1(rA_i)G + B_i$ for BV_i. Thus, for any two outgoing transactions, it is impossible to prove they were sent to the same person.
2) From the phase *Reward from CAG*, we know that $A = H_2(\mathrm{Sign}_z(m))$. Based on the security properties of hash function H_2, A's preimage $\mathrm{Sign}_z(m)$ is only known to CAG. Thus, by showing $\mathrm{Sign}_z(m)$, CAG can be regarded as the real signer. If BV_i still does not believe CAG is the real signer because A's preimage may be come from others, BV_i verifies $\mathrm{Sign}_z(m)$ by making use of Verify. Since (Sign, Verify) is secure digital signature pair, $\mathrm{Sign}_z(m)$ can be verifies by making use of CAG's public key in PKI. Because secure digital signature algorithm is unforgeable, the dispute is solved.
3) When BV suspects CAG has not rewarded it, it can query CAG by showing its receipt. By making use of CAG's help, BV can be ensured.

B. Performance Analysis

A practical BBARS scheme must be efficient in terms of computation cost and communication cost. In our BBARS scheme, two different PKCs are used: PKCs in PKI, PKCs on the blockchain. For PKCs on the blockchain, on the elliptic curve E, the scalar multiplication cost is denoted as $C_{E\mathrm{mul}}$, and the point addition cost is denoted as $C_{E\mathrm{add}}$. For PKCs in PKI, on the bilinear group pair $(\mathbb{G}_1, \mathbb{G}_2)$, the bilinear pairing cost is denoted as $C_{B\mathrm{pair}}$, the scalar multiplication cost is denoted as $C_{B\mathrm{mul}}$ and the point addition cost is denoted as $C_{B\mathrm{add}}$. Comparing to the above computation cost, the other computation cost is smaller. For the above denotations, Emul is the abbreviation of scalar multiplication, *E*add is the abbreviation of point addition on E. On the other hand, *B*pair is the abbreviation of bilinear pairing, *B*mul is the abbreviation of scalar multiplication, and *B*add is the abbreviation of point addition on $(\mathbb{G}_1, \mathbb{G}_2)$.

1) Theoretical Performance Analysis: For the performance of our scheme, the theoretical analysis is given in Table II. In the Table II, * denotes the entity does not take part in the procedure. We assume that the number of BVs is $|\mathbb{I}|$ and the number of LAGs is $|\mathbb{J}|$. *Small* denotes the computation cost is less than the five operations $C_{E\mathrm{mul}}, C_{E\mathrm{add}}, C_{B\mathrm{pair}}, C_{B\mathrm{mul}}$, and $C_{B\mathrm{add}}$. $n+1$ denotes the cardinality of the ring signature, i.e., the number of the possible signers. The procedure *Contract-based authorization* is performed between BV and CAG. BV's computation cost is

TABLE II
COMPUTATION COST OF THE DIFFERENT ENTITY

Procedure \ Entity	BV	LAG	CAG	Blockchain
Contract-based authorization	Small	*	$1C_{Bmul}$	*
Anonymous service and receipt	Small	$\frac{2}{\|\mathbb{I}\|}C_{Bpair}+2C_{Bmul}$ $+2(1-\frac{1}{\|\mathbb{I}\|})C_{Badd}$	*	*
Reward from CAG	*	*	$\|\mathbb{I}\|\|\mathbb{J}\|C_{Bpair}+2\|\mathbb{I}\|\|\mathbb{J}\|C_{Bmul}+$ $(\|\mathbb{I}\|\|\mathbb{J}\|-1)C_{Badd}+(2\|\mathbb{I}\|+2n$ $+5)C_{Emul}+(\|\mathbb{I}\|+3)C_{Eadd}$	*
Verification and Gain	$3C_{Emul}+1C_{Eadd}$	*	*	$4(n+1)C_{Emul}$ $+2(n+1)C_{Eadd}$

small and CAG's computation cost is $1C_{B\text{mul}}$. The procedure *Anonymous service and receipt* is performed between BV and LAG. BV's computation cost is small and CAG's computation cost is $(2/|\mathbb{I}|)C_{B\text{pair}}+2C_{B\text{mul}}+2(1-[1/|\mathbb{I}|])C_{B\text{add}}$ for each BV. The procedure *Reward from CAG* is performed by CAG. The computation cost is $(|\mathbb{I}||\mathbb{J}|C_{B\text{pair}}+2|\mathbb{I}||\mathbb{J}|C_{B\text{mul}})+(|\mathbb{I}||\mathbb{J}|-1)C_{B\text{add}}+(2|\mathbb{I}|+2n+5)C_{E\text{mul}}+(|\mathbb{I}|+3)C_{E\text{add}}$. The procedure *Reward from CAG* is performed by BV and blockchain. BV's computation cost is $3C_{E\text{mul}}+1C_{E\text{add}}$ and blockchain's computation cost is $4(n+1)C_{E\text{mul}}+2(n+1)C_{E\text{add}}$.

Currently, the PKCs on the blockchain is the elliptic curve over the finite field $\mathbb{F}_q$ where $|q|=256$ bits. In the bilinear group $(\mathbb{G}_1,\mathbb{G}_2)$, $\mathbb{G}_1$ is the supersingular elliptic curve over the finite field $\mathbb{F}_{\hat{q}}$ where $|\hat{q}|=512$ bits. According to the above two PKCs, we analyze our BBARS scheme's communication cost. In the procedure *Contract-based authorization*, CAG sends $c_{BV_i}=|\text{Cont}_i|+|\sigma_i|=1024+|\text{Cont}_i|$ bits to BV_i and sends $c_{LAG}=|\text{Tab}|$ bits to LAG. σ_i's size $|\sigma_i|$ is a constant 1024. Cont_i's size $|\text{Cont}_i|$ has the linear relation with the communication cost c_{BV_i}. At the same time, the communication cost c_{LAG} only comes from the size |Tab|. Thus, c_{LAG} has the linear relation with |Tab|. In the procedure *Anonymous service and receipt*, LAG_j sends $(m_{i,j},A_i,B_i,\bar{\sigma}_{i,j})$ to CAG and BV_i, respectively. The corresponding communication cost is $c_{\text{LAG}_j}=2(|m_{i,j}|+|A_i|+|B_i|+|\bar{\sigma}_{i,j}|)=2|m_{i,j}|+6144$ bits. For the communication cost c_{LAG_j}, $|A_i|$, $|B_i|$ and $|\bar{\sigma}_{i,j}|$ are the same constant 1024 bits. On the other hand, $|m_{i,j}|$ has the linear relation with c_{LAG_j}. In the procedure *Reward from CAG*, the final signature size is $c_{\text{Reward}}=|\sigma|=|I|+|A|+\sum_{i=0}^{n}|c_i|+\sum_{i=0}^{n}|r_i|=1024+506(n+1)$ bits. For the communication cost c_{Reward}, the sizes of I,A,c_i,r_i are different constants. They are 512 and 253 bits, respectively. At last, c_{Reward} has the linear relation with $n=|\mathcal{S}'|$.

2) Implementation: In order to demonstrate our BBARS scheme's effectiveness, we implemented it by making use of the simulation. We used the C programming language with the GMP (GMP-5.0.5), Miracl and PBC (pbc-0.5.13) libraries. In our implementation, both CAG and LAG ran on a laptop computer which has the features below.

1) *CPU:* Intel Core i7-3517U @ 1.90 GHz.
2) *Physical Memory:* 4 GB DDR3 1600 MHz.
3) *OS:* Ubuntu 13.04 Linux 3.8.0-19-generic SMP i686.

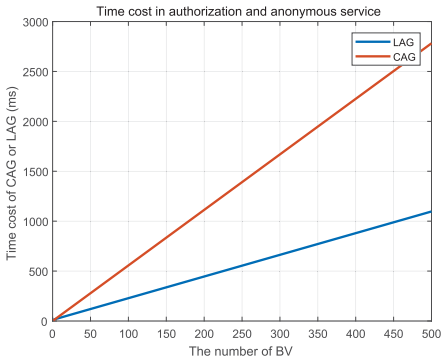


Fig. 3. CAG's time cost in *contract-based authorization* and LAG's time cost in *anonymous service and receipt*.

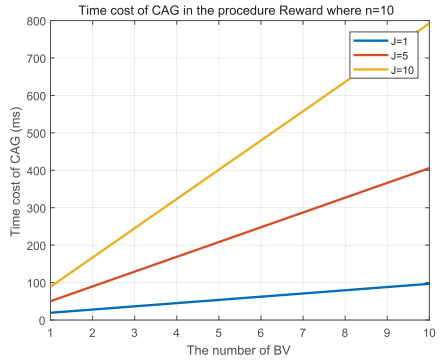


Fig. 4. CAG's time cost in *Reward from CAG*.

BV ran on a laptop which has the features below,

1) *CPU:* CPU I PDC E6700 3.2GHz.
2) *Physical Memory:* DDR3 2G.
3) *OS:* Ubuntu 11.10 over VMware-workstation-full-8.0.0.

Besides of the above simulation environment, we take use of the Monero blockchain and the corresponding PKCs. For the PKCs in PKI, we take use of the bilinear group $(\mathbb{G}_1,\mathbb{G}_2)$ where $\mathbb{G}_1$ is defined on the finite field $\mathbb{F}_{\hat{q}}$ with $|\hat{q}|=512$ bits. At the same time, $\mathbb{G}_1$ is a supersingular elliptic curve with 160-bit group order. Fig. 3 depicts the time cost of CAG and LAG in the procedures *Contract-based authorization* and *Anonymous service and receipt*, respectively. In the figure, x-axis represents the number of BV. The y-axis represents

TABLE III
COMMUNICATION COST (BITS)

Comm. \ Procedure	Contract-based authorization		Anonymous service and receipt	Reward($CAG \rightarrow Blockchain$)		
	$CAG \rightarrow BV_i$	$CAG \rightarrow LAG$	($LAG_j \rightarrow CAG$ (BV_i))	n=7	n=10	n=15
Comm. cost (bit)	3.15M	2.37M	46M	5.53M	6.5M	8.19 M

Notes: "Comm." is the abbreviation of "Communication".

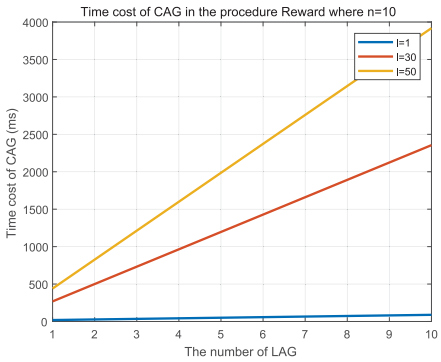


Fig. 5. CAG's time cost in *Reward from CAG*.

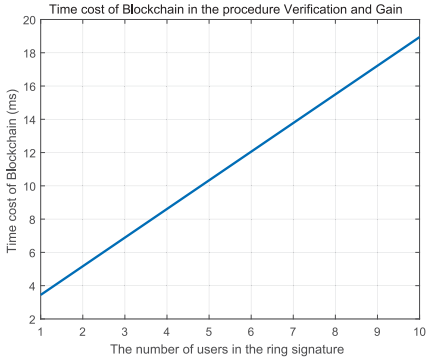


Fig. 6. Blockchain's time cost in *verification and gain*.

CAG and LAG's computing time in ms (i.e., milliseconds). Because CAG authorizes BV independently, CAG's time cost increases fastly along with the increasing of BV number. On the other hand, BV's authorization messages can be aggregated and verified by LAG. Thus, LAG's time cost increases slowly along with the increasing of BV number. Fig. 4 depicts the time cost of CAG in the procedures *Reward from CAG*. In the figure, *x*-axis represents the number of BV. The *y*-axis represents CAG's computing time in ms (i.e., milliseconds). In the figure, the number of the ring users is $n+1=11$. We implement the procedure for the different numbers $|\mathbb{J}|=1, |\mathbb{J}|=5, |\mathbb{J}|=10$. Fig. 5 also depicts the time cost of CAG in the procedures *Reward from CAG*. In the figure, *x*-axis represents the number of LAG. The *y*-axis represents CAG's computing time in ms. In the figure, the number of the ring users is $n+1=11$. We implement the procedure for the different numbers $|\mathbb{I}|=1, |\mathbb{I}|=30, |\mathbb{I}|=50$. From the Table II, we know that BV's time cost is constant. In our implementation, it is 1.29 ms. Fig. 6 depicts the time cost of blockchain in the procedures *Verification and Gain*. In the figure, *x*-axis represents the number of users in the ring signature. The *y*-axis represents blockchain's computing time in ms.

In order to show the intuition of communication cost, the experimental evaluation is given below. In the experiment, let the size of the message $m_{i,j}$ be 20M bits. The agreed contract Cont_i's is size is 2M bits. Suppose that Tab contains 5000 records. The other parameters and the implementation platform are the same as the computation prototype. The experimental results are given in the Table III. Based on the current communication technology, the communication cost is low. Thus, our scheme is efficient and practical.

VI. CONCLUSION

In this paper, we studied the privacy protection and precise reward architecture for V2G networks in the smart grid. By taking use of the properties of blockchain, we propose the novel concept of blockchain-based anonymous rewarding for V2G in the smart grid. The system model and security model are formalized. Based on the aggregated signature, ring signature, and blockchain, we design the first BBARS scheme. The analysis and implementation show that our BBARS scheme is provably secure and efficient.

In the future, we will further refine the system model and security model for different practical requirements. Based on the different practical requirements, we will propose the corresponding solutions. For example, how to avoid the misuse of contract-based authentication? Along with the rapid development of artificial intelligence (AI), it is also interesting and practical to study anonymous rewarding for AI-based V2G.

REFERENCES

[1] D. Wang, S. Saxena, J. Coignard, E. A. Iosifidou, and X. Guan, "Quantifying electric vehicle battery degradation from driving vs. V2G services," in *Proc. IEEE Power Energy Soc. Gen. Meeting (PESGM)*, 2016, pp. 1–5.

[2] (2007). *Car Prototype Generates Electricity and Cash*. [Online]. Available: http://www.sciencedaily.com/releases/2007/12/071203133532.htm

[3] S. Nakamoto. (2008). *Bitcoin: A Peer-to-Peer Electronic Cash System*. [Online]. Available: http://www.cryptovest.co.uk/resources/Bitcoin

[4] N. Bitansky, A. Chiesa, Y. Ishai, O. Paneth, and R. Ostrovsky, "Succinct non-interactive arguments via linear interactive proofs," in *Proc. TCC*, 2013, pp. 315–333.

[5] H. Lipmaa, "Succinct non-interactive zero knowledge arguments from span programs and linear error-correcting codes," in *Proc. Asiacrypt*, 2013, pp. 41–60.

[6] H. Wang, D. He, and Y. Ji, "Designated-verifier proof of assets for bitcoin exchange using elliptic curve cryptography," *Future Gener. Comput. Syst.*, Jul. 2017, doi: 10.1016/j.future.2017.06.028.

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

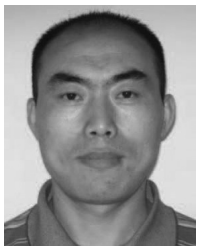
[8] H. Wang, B. Qin, Q. Wu, L. Xu, and J. Domingo-Ferrer, "TPP: Traceable privacy-preserving communication and precise reward for vehicle-to-grid networks in smart grids," *IEEE Trans. Inf. Forensics Security*, vol. 10, no. 11, pp. 2340–2351, Nov. 2015.

[9] W. Han and Y. Xiao, "Privacy preservation for V2G networks in smart grid: A survey," *Comput. Commun.*, vols. 91–92, pp. 17–28, Oct. 2016.

[10] Z. Wan, W. T. Zhu, and G. Wang, "PRAC: Efficient privacy protection for vehicle-to-grid communications in the smart grid," *Comput. Security*, vol. 62, pp. 246–256, Sep. 2016.
[11] X. Li and M. Sha, "Gauss factorials of polynomials over finite fields," *Int. J. Number Theory*, vol. 13, no. 8, pp. 2039–2054, 2017.
[12] S. Yang, X. Kong, and C. Tang, "A construction of linear codes and their complete weight enumerators," *Finite Fields Appl.*, vol. 48, pp. 196–226, Nov. 2017.
[13] S. Yang and Z.-A. Yao, "Complete weight enumerators of a class of linear codes," *Discr. Math.*, vol. 340, no. 4, pp. 729–739, 2017.
[14] S. Yang, Z.-A. Yao, and C.-A. Zhao, "The weight distributions of two classes of p-ary cyclic codes with few weights," *Finite Fields Appl.*, vol. 44, no. 8, 2017, pp. 76–91.
[15] W. Lv and F. Wang, "Adaptive tracking control for a class of uncertain nonlinear systems with infinite number of actuator failures using neural networks," *Adv. Difference Equ.*, vol. 2017, p. 374, 2017.
[16] L. Li, Z. Wang, Y. Li, H. Shen, and J. Lu, "Hopf bifurcation analysis of a complex-valued neural network model with discrete and distributed delays," *Appl. Math. Comput.*, vol. 330, pp. 152–169, Aug. 2018.
[17] Y. Zhang, M. Zhao, J. Su, X. Lu, and K. Lv, "Novel model for cascading failure based on degree strength and its application in directed gene logic networks," *Comput. Math. Methods Med.*, vol. 2018, pp. 1–9, Feb. 2018.
[18] F. Gao *et al.*, "A blockchain-based privacy-preserving payment mechanism for vehicle-to-grid networks," *IEEE Netw.*, vol. 32, no. 6, pp. 184–192, Nov./Dec. 2018, doi: 10.1109/MNET.2018.1700269.
[19] A. Dorri, M. Steger, S. S. Kanhere, and R. Jurdak, "BlockChain: A distributed solution to automotive security and privacy," *IEEE Commun. Mag.*, vol. 55, no. 12, pp. 119–125, Dec. 2017.
[20] L. Li *et al.*, "CreditCoin: A privacy-preserving blockchain-based incentive announcement network for communications of smart vehicles," *IEEE Trans. Intell. Transp. Syst.*, vol. 19, no. 7, pp. 2204–2220, Jul. 2018, doi: 10.1109/TITS.2017.2777990.
[21] G. Liang, S. R. Weller, F. Luo, J. Zhao, and Z. Y. Dong, "Distributed blockchain-based data protection framework for modern power systems against cyber attacks," *IEEE Trans. Smart Grid*, to be published, doi: 10.1109/TSG.2018.2819663.
[22] Z. Li *et al.*, "Consortium blockchain for secure energy trading in industrial Internet of Things," *IEEE Trans. Ind. Informat.*, vol. 14, no. 8, pp. 3690–3700, Aug. 2018, doi: 10.1109/TII.2017.2786307.
[23] N. Z. Aitzhan and D. Svetinovic, "Security and privacy in decentralized energy trading through multi-signatures, blockchain and anonymous messaging streams," *IEEE Trans. Depend. Secure Comput.*, vol. 15, no. 5, pp. 840–852, Sep./Oct. 2018.
[24] N. Koblitz, A. Menezes, and S. Vanstone, "The state of elliptic curve cryptography," *Designs Codes Cryptography*, vol. 19, nos. 2–3, pp. 173–193, 2000.
[25] H. Wang, D. He, and J. Han, "VOD-ADAC: Anonymous distributed fine-grained access control protocol with verifiable outsourced decryption in public cloud," *IEEE Trans. Services Comput.*, to be published, doi: 10.1109/TSC.2017.2687459.
[26] D. Boneh and M. Franklin, "Identity-based encryption from the weil pairing," in *Proc. CRYPTO*, 2001, pp. 213–229.
[27] A. Miyaji, M. Nakabayashi, and S. Takano, "New explicit conditions of elliptic curve traces for FR-reduction," *IEICE Trans. Fundam.*, vol. 5, pp. 1234–1243, 2001.
[28] D. Boneh, B. Lynn, and H. Shacham, "Short signatures from the weil pairing," in *Proc. Asiacrypt*, 2001, pp. 514–532.
[29] D. Boneh, C. Gentry, B. Lynn, and H. Shacham, "Aggregate and verifiably encrypted signatures from bilinear maps," in *Proc. Eurocrypt*, 2003, pp. 416–432.
[30] H. Wang, Z. Wang, and J. Domingo-Ferrer, "Anonymous and secure aggregation scheme in fog-based public cloud computing," *Future Gener. Comput. Syst*, vol. 78, pp. 712–719, Jan. 2018.
[31] R. L. Rivest, A. Shamir, and Y. Tauman, "How to leak a secret," in *Proc. Asiacrypt*, 2001, pp. 552–565.
[32] N. Saberhagen. *CryptoNote Version 2.0*. Accessed: Jan. 11, 2019. [Online]. Available: https://static.coinpaprika.com/storage/cdn/whitepapers/1611.pdf
[33] H. Wang, D. He, J. Yu, and Z. Wang, "Incentive and unconditionally anonymous identity-based public provable data possession," *IEEE Trans. Services Comput.*, to be published, doi: 10.1109/TSC.2016.2633260.
[34] H. Wang, D. He, and J. Yu, "Privacy-preserving incentive and rewarding scheme for crowd computing in social media," *Inf. Sci.*, vol. 470, pp. 15–27, Jan. 2019.
[35] A. Kumar, C. Fischer, S. Tople, and P. Saxena, "A traceability analysis of Monero's blockchain," in *Proc. ESORICS*, 2017, pp. 153–173.

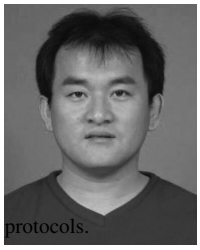
Huaqun Wang received the B.S. degree in mathematics education from Shandong Normal University, Jinan, China, in 1997, the M.S. degree in applied mathematics from East China Normal University, Shanghai, China, in 2000, and the Ph.D. degree in cryptography from the Nanjing University of Posts and Telecommunications, Nanjing, China, in 2006.

He is currently a Professor with the Nanjing University of Posts and Telecommunications. His current research interests include applied cryptography, network security, and cloud computing security.



Qihua Wang was born in Weifang, China, in 1979. He received the B.S. degree in computer science and technology from Shandong Normal University, Jinan, China, in 2003, the M.S. degree in computer science and technology from the Harbin University of Science and Technology, Harbin, China, in 2008, and the Ph.D. degree in control theory and control engineering from Dalian Maritime University, Dalian, China, in 2015.

He is currently a Teacher with Jining Medical University, Jining, China, and a Post-Doctoral Researcher with the University of Electronic Science and Technology of China, Chengdu, China. His current research interests include applied cryptography, Internet of Things, network security, and network location.



Debiao He received the Ph.D. degree in applied mathematics from the School of Mathematics and Statistics, Wuhan University, Wuhan, China, in 2009.

He is currently an Associate Professor with the State Key Laboratory of Software Engineering, Computer School, Wuhan University. His current research interest includes cryptography and information security, in particular, cryptographic protocols.

Qi Li received the Ph.D. degree in computer system architecture from Xidian University, Xi'an, China, in 2014.

He is a Lecturer with the School of Computer Science, Nanjing University of Posts and Telecommunications, Nanjing, China. His current research interests include cloud security, information security, and applied cryptography.

Zhe Liu received the B.S. and M.S. degrees from Shandong University, Jinan, China, in 2008 and 2011, respectively, and the Ph.D. degree from the Laboratory of Algorithmics, Cryptology and Security, University of Luxembourg, Luxembourg City, Luxembourg, in 2015.

He was a Researcher with SnT, University of Luxembourg. He is a Professor with the College of Computer Science and Technology, Nanjing University of Aeronautics and Astronautics, Nanjing, China. He has co-authored over 70 research peer-reviewed journal and conference papers. His current research interests include computer arithmetic and information security.

Dr. Liu was a recipient of the Prestigious FNR Awards 2016—Outstanding Ph.D. Thesis Award for his contributions in cryptographic engineering on IoT devices.