An Efficient Multilayered Linkable Ring Signature Scheme With Logarithmic Size for Anonymous Payment in Vehicle-to-Grid Networks

Yulin Liu, Debiao He, *Member, IEEE*, Zijian Bao, Huaqun Wang, Muhammad Khurram Khan, and Kim-Kwang Raymond Choo, *Senior Member, IEEE*

***Abstract*—Vehicle-to-Grid (V2G) networks are potential solutions to addressing energy and environmental challenges, although security remains a key concern. For example, attackers may seek to obtain private information from frequent electricity/service exchanges between electric vehicles (EVs) and smart grids (SG) in V2G networks. While there have been successful attempts in using ring signatures to achieve privacy-preserving payment and ensure reliable services in V2G many existing ring signature-based payment proposals has significant signature size. To address the aforementioned problem, we propose Emularis, an efficient multilayered linkable ring signature scheme with a logarithmic size. We also implement an anonymous payment scheme for V2G using Emularis. We then prove that Emularis guarantees security and privacy requirements through rigorous security analysis. Furthermore, our scheme significantly outperforms existing schemes in terms of communication and computation costs. Extensive experimental results indicate that our scheme is suitable for the deployment in V2G security-related applications.**

***Index Terms*—Anonymous payment, privacy preservation, ring signature, V2G networks.**

Manuscript received 30 August 2022; revised 12 October 2022; accepted 20 October 2022. Date of publication 25 October 2022; date of current version 19 May 2023. The work was supported in part by the National Key Research and Development Program of China under Grant 2021YFA1000600, in part by the National Natural Science Foundation of China under Grants U21A20466, 62172307, and 62272238, in part by the Shandong Provincial Key Research and Development Program under Grants 2020CXGC010107 and 2021CXGC010107, in part by the Special Project on Science and Technology Program of Hubei Provience under Grant 2020AEA013, in part by the Natural Science Foundation of Hubei Province under Grant 2020CFA052, and in part by the Wuhan Municipal Science and Technology Project under Grant 2020010601012187. The work of Muhammad Khurram Khan is supported by King Saud University, Riyadh, Saudi Arabia under project number RSP-2022/12. The work of Kim-Kwang Raymond Choo was supported only by the Cloud Technology Endowed Professorship. *(Corresponding authors: Debiao He; Zijian Bao.)*

Yulin Liu is with the School of Cyber Science and Engineering, Wuhan University, Wuhan 430072, China, and also with the Shandong Provincial Key Laboratory of Computer Networks, Qilu University of Technology (Shandong Academy of Sciences), Jinan 250014, China (e-mail: liuyulin@whu.edu.cn).

Debiao He is with the School of Cyber Science and Engineering, Wuhan University, Wuhan 430072, China, and also with the Shanghai Key Laboratory of Privacy- Preserving Computation, MatrixElements Technologies, Shanghai 201204, China (e-mail: hedebiao@163.com).

Zijian Bao is with the School of Cyber Science and Engineering, Wuhan University, Wuhan 430072, China (e-mail: baozijian@whu.edu.cn).

Huaqun Wang is with the Jiangsu Key Laboratory of Big Data Security and Intelligent Processing, College of Computer, Nanjing University of Posts and Telecommunications, Nanjing 210023, China (e-mail: wanghuaqun@aliyun.com).

Muhammad Khurram Khan is with the Center of Excellence in Information Assurance (CoEIA), King Saud University, Riyadh 11564, Saudi Arabia (e-mail: mkhurram@ksu.edu.sa).

Kim-Kwang Raymond Choo is with the Department of Information Systems and Cyber Security, University of Texas at San Antonio, San Antonio, TX 78249 USA (e-mail: raymond.choo@fulbrightmail.org).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TIV.2022.3216949.

Digital Object Identifier 10.1109/TIV.2022.3216949

I. Introduction

LONG-term rises in gasoline prices and the harm of global warming have provided the automotive industry with a new competitive environment [1]. Electric vehicle (EV) technology has steadily become a hot spot of global research and attention to ease these problems. Once these new, innovative vehicles have been fully charged, they can travel long distances at a moderate speed. Besides, EVs have numerous benefits, such as being environmentally friendly, energy-saving, and easily accessible, which makes the adoption of EVs quite appealing. It is predicted that by 2035, the global market size of intelligent networked vehicles will reach trillions.

The V2G network, as a critical component of smart grids (SG), represents a new direction for the future development of EVs [2], [3], [4]. Fig. 1 depicts the typical network structure of V2G, which consists of five parts, including EV, Charging Station (CS), Aggregator, Smart Grid Control Center (SGCC), and the Internet. Each aggregator in a different region connects to several neighboring CSs. EVs send their relevant information, such as identity, battery status, and charging data, to CSs. The aggregator collects and verifies the information stored in CSs to monitor EVs. Then, aggregators communicate with SGCC via the wired or wireless Internet. SGCC takes charge of some management work (*e.g.*, calculating the total price of electricity according to the charging and discharging demand, and scheduling the entire V2G network) [5].

The bidirectional power transmission between EVs and SG generates a great deal of payment information data on electricity usage in V2G [6]. As shown in Fig. 1, local aggregators distribute electricity from SGCC and control several CSs to charge EVs. When an EV wants to charge, it needs to pass the authentication of a local aggregator to get charging services. Nevertheless, during this interaction process involving payment data, malicious criminals can attack CSs or local aggregators to seek EV users' charging records, thereby obtaining sensitive

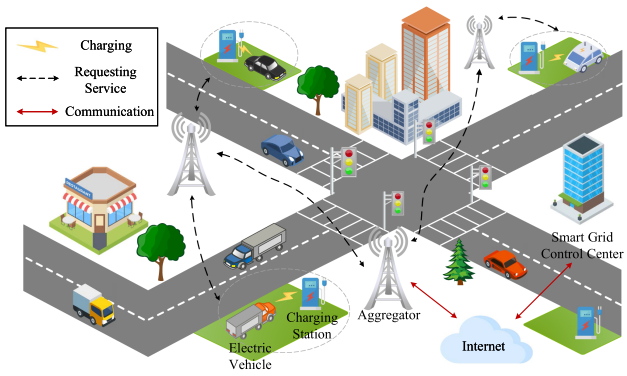


Fig. 1. Typical network structure of V2G.

information (such as identities, living habits, locations, and action trajectory) [7]. If the above-mentioned sensitive information is utilized maliciously, violent situations such as robbery and theft may occur, putting the safety of EV owners at risk. As a result, V2G networks require a reliable, privacy-preserving, and available payment scheme to adapt to charging / discharging transactions of many EVs.

Generally speaking, ring signature [8], an emerging digital signature technology, can not only ensure authentication but also provide anonymity for the users. In a ring signature, a signer selects public keys of various ring members at random to form an ad-hoc group, then combines their public keys, private keys, and random numbers to complete the signature. Verifiers can convince that the signature came from a member of the group, without necessarily confirming the signer's specific identity. The complete anonymity and unforgeability of the ring signature make it considerable attention. As of late, plenty of scholars have proposed many applications of ring signatures in vehicular networks [9], [10], [11], [12].

Nevertheless, ring signatures provide unconditional anonymity, which may be too strong in some scenarios, *e.g.*, e-voting, and e-cash. Therefore, Liu et al. presented [13] linkable spontaneous anonymous group signature (LSAG), a linkable ring signature (LRS) protocol that can detect whether two signatures were generated using the same private key (while a verifier still has no idea about the signer's actual identity). LRS provides linkable anonymity instead of strong unconditional anonymity. It can guarantee privacy protection in cryptocurrency applications while also resisting double-spending attacks.

Monero [14], a well-known open-source cryptocurrency based on the CryptoNote protocol [15], leverages three technologies of one-time addresses, LRS, and ring confidential transactions (RingCT) [16] to provide robust blockchain anonymity. The multilayered linkable spontaneous anonymous group signatures (MLSAG) scheme, which supports multiple inputs, is a variant of the LRS. In Monero, the orchestration of the MLSAG scheme and the RingCT protocol can obscure the transaction amount, prevent double-spending attacks and further enhance privacy-preserving in payment scenarios [16].

Although there exist some anonymous mechanisms based on ring signature protocol, the drawback that a signature size scales linearly with the number of ring members tremendously hinders the application in transaction-intensive scenarios, *e.g.*, V2G networks. The large communication overhead leads to high system latency and low throughput. This results in the ring size being limited to a very small number of members, thus drastically limiting the anonymity level.

Therefore, recent work strives to mitigate the size restriction of ring signatures. There are two main cryptographic primitives for building ring signatures of smaller size, accumulator and zero-knowledge proof. In accumulator-based constructions like [17], [18], the signature is constant-size, independent of the number of ring members. However, their constructions require a trusted setup, which may often not be preferable. Zero-knowledge proof-based protocols [19], [20], [21], [22], [23], [24] seem to be desirable since most of them purposed the state-of-the-art logarithmic size for utility.

A. Contributions

We introduce a communication-efficient Multilayered Linkable Ring Signature, called *Emularis*, in which the signature size increases logarithmically with the number of ring members. Furthermore, we implement an anonymous payment scheme for V2G by using the proposed Emularis. In a bit more detail, the core contributions of this work are threefold as follows.

- First, we construct a new multilayered linkable ring signature protocol based on MLSAG for achieving privacy-preserving payment in V2G networks. Our scheme not only guarantees the anonymity requirement of the user's identity but also yields a small signature whose size grows logarithmically as the size of ring increases.
- Second, we prove that our proposed multilayered ring signature protocol fulfills the security and privacy requirements, provided that the underlying mathematical assumption holds.
- Finally, we implement Emularis scheme and conduct extensive simulations to evaluate its performance. The experimental results illustrate that our scheme is feasible and effective for the deployment in V2G privacy-friendly applications compared to the existing works.

B. Paper Outline

The remainder of our paper is arranged as follows. We concisely review the relevant literature on the privacy preservation of V2G networks in Section II. Section III describes the necessary preliminary knowledge. The design details (along with the security analysis and performance evaluation) of Emularis and anonymous payment scheme are illustrated in Sections IV and V, respectively. At last, we summarize the entire work of our paper in Section VI.

II. Related Work

A large number of works on the privacy-preserving cryptographic approach have been put forward for V2G networks, mainly involving anonymous authentication, identity / location privacy, privacy-preserving payment, and so on.

The first V2G privacy-considered scheme P^2, designed by Yang et al., with precise reward system architecture and privacy-preserving communication [25] built upon an identity-based restrictive partially blind signature to protect the privacy of EVs. Unfortunately, the certificates generated by this scheme are easily forged by adversaries. Wang et al. [26] discovered the security flaws in [25] and proposed a new V2G networks privacy protection scheme that utilized bilinear pairing and identity-based restrictive partially blind signature techniques. Nevertheless, this scheme has heavy computation and communication overheads. In [10], [27], [28], the authors proposed various privacy-preserving authentication protocols to achieve secure communication in V2G networks. A survey [29] enumerated a variety of privacy preservation concerns and proposals in V2G networks, along with existing privacy preservation and possible solutions to unresolved issues.

The frequent exchange of payment information between EVs and SG may expose sensitive information, for instance, the identity / location privacy of EVs. Very recently, various anonymous payment mechanisms have been proposed to support privacy protection for V2G. Liu et al. [30] enhanced the location privacy of vehicles with a privacy-preserving solution that uses an anonymous payment system. Later, Au et al. [31] improved the payment system protocol in [30], which employs BBS+ signature and zero-knowledge proof. This solution can not only achieve the location privacy preservation of the vehicle, but also support the tracking of a vehicle's location in the event that the vehicle is stolen.

Saxena et al. [32] presented the network security / privacy requirements and challenges in V2G networks. This article demonstrates that a privacy-preserving payment system for anonymously handling plenty of frequent transactions is in urgent need. In [6], the authors presented a privacy-preserving payment protocol based on blockchain to resolve the conflicts between data sharing and privacy protection in V2G networks. Similarly, Wan et al. [33] designed a blockchain-based fair exchange scheme for V2G that protects privacy using zk-SNARK to concurrently accomplish fairness and privacy.

With the prevalence of ring signatures, several works utilized them to achieve privacy-preserving payment. Monero [14] uses ring signatures to hide the real spender of a transaction from several irrelevant users and utilizes the RingCT protocol to obscure the transaction amount. However, the large signature size of ring signatures originally used in Monero greatly expands the transaction size and limits the speed of transaction processing. Wang et al. [11] utilized Monero as the underlying cryptocurrency to achieve anonymous rewarding payments for V2G, in which the linkable ring signature used also faces the shortcoming of big signature size.

Reducing the size of ring signatures has therefore attracted numerous researchers. Several accumulator-based ring signature protocols [17], [18] have been proposed to achieve the constant-size signature. Unfortunately, these schemes required undesirable trusted setup assumptions. Another mainstream approach to constructing communication-efficient ring signatures is to take advantage of zero-knowledge proof [19], [20], [21], [22], [23], [24]. Yuen et al. [24] built DualRing, a novel generic ring signature structure that enables the signature size to be shortened to a logarithmic size by leveraging a new proof system. Inspired by DualRing [24], we shorten the signature size of MLSAG [16] from linear to logarithmic in the number of the ring.

From the literature survey, we note that most of existing privacy-preserving payment proposals cannot work effectively for V2G networks in practical applications. They either require a trusted setup or cost much to preserve EVs' privacy.

TABLE I
NOTATIONS AND DESCRIPTIONS

Notations	Descriptions
λ	A security parameter
$y \leftarrow \mathsf{ALG}(x)$	The procedure of running the randomized algorithm ALG on input x and output y
q	A large prime
$\mathbb{G}$	An additive cyclic group of prime order q
G	The generator of additive cyclic group $\mathbb{G}$
$\mathcal{H}^s(\cdot)$	The hash function of $\{0,1\}^* \rightarrow \mathbb{Z}_q$
$\mathbb{Z}_q$	The set of integers $\{0,1,2,...,q-1\}$
$\mathcal{H}^p(\cdot)$	The hash function of $\{0,1\}^* \rightarrow \mathbb{G}$
uP	Point multiplication in additive cyclic group $\mathbb{G}$
$[a]$	The set of integers$\{1, ..., a\}$
$\{t_i\}_{i=1}^m$	The array of $\{t_1, ..., t_m\}$
n	The number of ring members
m	The number of accounts (public/private keys) of each member
$\left(\vec{pk}_i, \vec{sk}_i\right)$	The private/public key-vector pairs of the ith ring member
$\mathcal{L}_{pk}$	The public key set
ACT	The account set
ξ	The signer's secret index
I	The key image
σ	The Emularis signature of message
$\mathcal{PPT}$	The abbreviation of probabilistic polynomial time
NISA	The abbreviation of non-interactive sum argument

III. PRELIMINARY KNOWLEDGE

A. Cryptographic Primitives

This part begins with an overview of the main notations that this paper uses, followed by a succinct presentation of MLSAG scheme in RingCT, sum argument of knowledge, and Pedersen commitment.

1) Notations: Throughout the paper, we present the main notations in Table I, together with their descriptions. Then, a mathematical assumption called Discrete Logarithm Problem (DLP) is defined as follows:

Definition 1 (Discrete Logarithm Problem): Given any two elements $P \in \mathbb{G}, Q \in \mathbb{G}$, it is hard for a $\mathcal{PPT}$ algorithm to find a integer $x \in \mathbb{Z}_q$ that satisfies $Q = xP$.

2) Multilayered Linkable Spontaneous Anonymous Group Signatures: Our Emularis is designed based on the linkable ring signature used by the RingCT protocol in Monero. The MLSAG sheme [16] allows for addressing multi-input needs in the cryptocurrency scenario. In contrast to LSAG [13], MLSAG uses a set of n key-vectors rather than n keys to generate linkable ring signatures, which makes applying

multiple inputs and outputs possible. Put simply, one has to sign multi-input transactions with m private keys. MLSAG signatures scheme is a composition of five algorithms $\Pi = (\texttt{Setup}, \texttt{KeyGen}, \texttt{Sign}, \texttt{Verify}, \texttt{Link})$, the details of which are defined such that:

- $\mathsf{pp} \leftarrow \mathsf{Setup}(\lambda)$: On input a security parameter λ, returns public system parameters pp.
- $(\vec{pk}_i, \vec{sk}_i) \leftarrow \mathsf{KeyGen}(\lambda)$: On input a security parameter λ, returns user$_i$'s public–private key pairs $(\vec{pk}_i, \vec{sk}_i)$ of length m, where $\vec{pk}_i = (pk_i^1, \ldots, pk_i^m)$, $\vec{sk}_i = (sk_i^1, \ldots, sk_i^m)$, $i \in [max]$.
- $\sigma \leftarrow \mathsf{Sign}(\mathcal{L}_{pk}, M, \vec{sk}_i)$: On input a message M, a group of public keys $\mathcal{L}_{pk} \subseteq S = (\vec{pk}_1, \ldots, \vec{pk}_{max})$ and the signer's private key-vectors $\vec{sk}_i$ with corresponding public key-vectors $\vec{pk}_i \in \mathcal{L}_{pk}$, the algorithm returns a signature σ over M.
- $0/1 \leftarrow \mathsf{Verify}(\mathcal{L}_{pk}, M, \sigma)$: On input a message M, a group of public keys $\mathcal{L}_{pk} \subseteq S$ and a signature σ, the algorithm verifies the validity of σ. If valid, the algorithm returns 1. Else, it returns 0.
- $linked/unlinked \leftarrow \mathsf{Link}(M, M^*, \sigma, I, \sigma^*, I^*)$: On input two different messages M, M^*, and their corresponding linkable ring signatures (σ, I), (σ^*, I^*), the algorithm checks if $I = I^*$. If σ, σ^* are valid and the equation $I = I^*$ holds, the algorithm outputs $linked$, namely, the two signatures σ, σ^* were signed by the same private key. Else, it returns $unlinked$.

3) Sum Arguments of Knowledge: Yuen et al. [24] proposed a non-interactive argument of knowledge called Sum Argument (NISA) based on Bulletproofs [34]. NISA is a sum argument proving the following relation:

$$\left\{\left(\vec{Q} \in \mathbb{G}^n, P \in \mathbb{G}, c \in \mathbb{Z}_q; \vec{a} \in \mathbb{Z}_q^n\right) : P = \vec{a} \cdot \vec{Q} \wedge c = \sum \vec{a}\right\}$$

where $\vec{Q}$ represents $(Q_1, \ldots, Q_n)$, $\vec{a}$ represents $(a_1, \ldots, a_n)$, $\vec{a} \cdot \vec{Q}$ denotes $\sum_{i=1}^{n} a_i Q_i$. The sum argument is a variation of the inner product argument used in Bulletproofs, which makes a verifier believe that the prover has the knowledge of $\vec{a}$, such that $P = \vec{a} \cdot \vec{Q}$ and $c = \sum \vec{a}$. It is worth noting that NISA is much simpler and has lower computational overhead than Bulletproofs.

On the whole, NISA consists of two algorithms (Proof, Verify) such that:

- $\pi \leftarrow \mathsf{NISA.Proof}(\{param, \vec{Q}, P, c\}, \vec{a})$ is a $\mathcal{PPT}$ algorithm. It takes a tuple of parameters $param$, $\vec{Q} \in \mathbb{G}^n$, $P \in \mathbb{G}$, $c \in \mathbb{Z}_q$, a verctor $\vec{a}$, and returns a proof π.
- $0/1 \leftarrow \mathsf{NISA.Verify}(param, \vec{Q}, P, c, \pi)$ is a deterministic algorithm run by the verifier. Its input is a tuple of parameters $param$, $\vec{Q} \in \mathbb{G}^n$, $P \in \mathbb{G}$, $c \in \mathbb{Z}_q$, and a proof π. If π is valid, it returns 1. Else, returns 0.

4) Pedersen Commitment: The commitment scheme allows an entity to commit a selected value while hiding the value from others. Pedersen commitment [35] is a commonly used homomorphic commitment scheme, which is easy to implement an elliptic curve with a base point G.

$\mathsf{CMT}(r, v) = r \cdot G + v \cdot W$ is a commitment to a secret value v with randomness r, where W is another base point of the elliptic curve (note that the discrete logarithm relationship between W and G is unknown). The following holds trivially for the property of homomorphic:

$$\mathsf{CMT}(r_0, v_0) + \mathsf{CMT}(r_1, v_1) = \mathsf{CMT}(r_0 + r_1, v_0 + v_1)$$

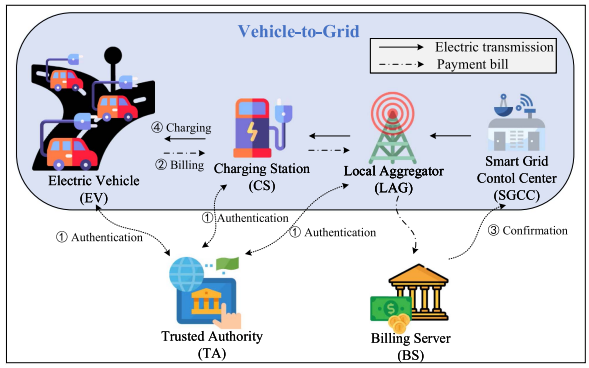


Fig. 2. System model of charging/payment scenarios in V2G networks.

B. System Model

We give a general system model of charging / payment scenarios in V2G networks, as shown in Fig. 2. Our scheme contains six various entities, *i.e.*, Trusted Authority (TA), Electric Vehicles (EVs), Charging Stations (CSs), Local Aggregator (LAG), Smart Grid Control Center (SGCC), and Billing Server (BS). Their functions are outlined as follows.

- *TA:* TA is a completely trusted third-party organization that is mainly in charge of the security foundation of the entire system. TA can obtain detailed information data to support billing services.
- *EV:* EVs travel in geographically dispersed locations, and they access V2G networks by authenticating to the TA. After an EV initiates a charging request, it needs to pay the electricity bill to BS anonymously, and then SGCC assigns the electricity from SG to the EV.
- *CS:* A CS is responsible for the data exchange between EVs and LAG. After first matching the demands of EVs and receiving the authentication confirmation from LAG, it can provide charging services for EVs.
- *LAG:* A LAG typically acts as a power and wireless communication service access point for EVs. LAGs are deployed in local areas (*e.g.*, public parking lots) to assist EVs and SGCC with information matching and secure communications, such as monitoring the state of charge (SoC) of EVs' real-time batteries, forwarding the current electricity price from the SGCC to EVs, and controlling charging / discharging operations for EVs.
- *SGCC:* SGCC is responsible for accurately managing the power grid. According to the collected charging demands from LAG, SGCC calculates the total amount / price of trading electricity. Moreover, it schedules the entire V2G network.
- *BS:* Every participant owns several accounts from power grids and deposits some money into these accounts. The BS acts as a bank responsible for billing and electricity price update operations.

C. Security Model

Subsequently, we take EVs' privacy information into account in this paper. Our Emularis scheme must satisfy the following security attributes in a real-world environment:

- Unforgeability: The legal EVs can only generate valid signatures for their own transaction information. Any other EVs cannot impersonate him/her to provide a valid signature that can pass the LAG's verification.
- Anonymity: EV remains anonymous in the process of payment, that is, the real identity cannot be identified by any other entity. The platform cannot directly obtain users' personal information, which can strongly guarantee the privacy protection of users' core information.
- Linkability: It can be detected if a malicious EV signs a transaction with the same private key. In other words, the signature will be linked if he/she spends a coin twice or more, thus preventing double-spending attacks.
- Non-frameability: The malicious attacker cannot generate the same key image as the real EV, and this property ensures that no other entity can frame an honest EV.

Without loss of generality, we refine some oracles used in the formal security analysis. We denote the adversary by $\mathcal{A}$ and the simulator by $\mathcal{S}$.

- Random Oracle $(r \leftarrow \mathcal{RO}(i))$: It takes a value i as input and returns a random number r.
- Corruption Oracle $(sk_i \leftarrow \mathcal{CO}(pk_i))$: It takes a public key pk_i as input, and then the associated private key sk_i is returned.
- Signing Oracle $(\sigma \leftarrow \mathcal{SO}(\mathcal{L}_{pk}, pk_i, M))$: It takes a public key set $\mathcal{L}_{pk} = (pk_1, \ldots, pk_n)$, a signer's public key $pk_i (i \in [n])$ and a message M as input, and generates a valid signature σ as output.

Herein, we give a formal security definition of Emularis scheme as described below.

Definition 2 (Unforgeability): A following game $\mathsf{Game}_{\mathsf{UNF}}$ between a $\mathcal{PPT}$ adversary $\mathcal{A}$ and the simulator $\mathcal{S}$ defines the unforgeability of Emularis:

- $\mathcal{S}$ generates the public system parameters pp for $\mathcal{A}$.
- The oracles $\mathcal{CO}, \mathcal{SO}$ and $\mathcal{RO}$ are adaptively accessible to $\mathcal{A}$.
- $\mathcal{A}$ forges a valid signature σ^* on a message M^* and a public key set $\mathcal{L}_{pk}$.

We say that $\mathcal{A}$ wins $\mathsf{Game}_{\mathsf{UNF}}$ if:

1) σ^* is a valid signature for message M^*.
2) All the public keys in $\mathcal{L}_{pk}$ are not corrupted, that is, $\mathcal{A}$ doesn't get the private key of any ring member in $\mathcal{L}_{pk}$ through the $\mathcal{CO}$.
3) σ^* is not queried to the signature oracle $\mathcal{SO}$.

Emularis scheme is unforgeable if $\mathsf{Adv}_{\mathcal{A}}^{\mathsf{UNF}}$ is negligible where $\mathsf{Adv}_{\mathcal{A}}^{\mathsf{UNF}}$ is the advantage of $\mathcal{A}$ winning $\mathsf{Game}_{\mathsf{UNF}}$.

Definition 3 (Anonymity): A following game $\mathsf{Game}_{\mathsf{ANO}}$ between a $\mathcal{PPT}$ adversary $\mathcal{A}$ having limitless computational power and the simulator $\mathcal{S}$ defines the anonymity of Emularis:

- $\mathcal{S}$ generates the public system parameters pp for $\mathcal{A}$.
- The oracles $\mathcal{RO}, \mathcal{SO}$ are adaptively accessible to $\mathcal{A}$.
- $\mathcal{A}$ gives a tuple $(M^*, \mathcal{L}_{pk}, i_0, i_1)$ to $\mathcal{S}$, where $\vec{pk}_{i_0}, \vec{pk}_{i_1} \in \mathcal{L}_{pk}$.
- $\mathcal{S}$ randomly selects a bit $b \in \{0, 1\}$ and then returns σ_{i_b} to $\mathcal{A}$, where σ_{i_b} is generated using the private key $\vec{sk}_{i_b}$.
- $\mathcal{A}$ returns a bit $b' \in \{0, 1\}$.

We say that $\mathcal{A}$ wins $\mathsf{Game}_{\mathsf{ANO}}$ if $b' = b$.

Emularis scheme is anonymous if $\mathsf{Adv}_{\mathcal{A}}^{\mathsf{ANO}}$ is negligible where $\mathsf{Adv}_{\mathcal{A}}^{\mathsf{ANO}}$ is the advantage of $\mathcal{A}$ winning $\mathsf{Game}_{\mathsf{ANO}}$.

Definition 4 (Linkability): A following game $\mathsf{Game}_{\mathsf{LINK}}$ between a $\mathcal{PPT}$ adversary $\mathcal{A}$ and the simulator $\mathcal{S}$ defines the linkability of Emularis:

- $\mathcal{S}$ generates the public system parameters pp for $\mathcal{A}$.
- The oracles $\mathcal{CO}, \mathcal{SO}$ are adaptively accessible to $\mathcal{A}$.
- $\mathcal{A}$ returns two tuples $(\mathcal{L}_{pk0}, M_0^*, \sigma_0^*)$ and $(\mathcal{L}_{pk1}, M_1^*, \sigma_1^*)$.

We say that $\mathcal{A}$ wins $\mathsf{Game}_{\mathsf{LINK}}$ if:

1) $\mathcal{A}$ can obtain only one private key sk_i corresponding to the public key pk_i from the oracle $\mathcal{CO}$.
2) σ_0^*, σ_1^* are both valid signatures signed regarding public key set $\mathcal{L}_{pk0}$ and $\mathcal{L}_{pk1}$, respectively, and there is a common public key $\hat{pk}$ in both $\vec{pk}^* \in \mathcal{L}_{pk0}$ and $\vec{pk}' \in \mathcal{L}_{pk1}$. In particular, they are not the outputs of the oracle $\mathcal{SO}$.
3) The two signatures σ_0^*, σ_1^* are unlinked.

Emularis scheme is linkable if $\mathsf{Adv}_{\mathcal{A}}^{\mathsf{LINK}}$ is negligible where $\mathsf{Adv}_{\mathcal{A}}^{\mathsf{LINK}}$ is the advantage of $\mathcal{A}$ winning $\mathsf{Game}_{\mathsf{LINK}}$.

Definition 5 (Non-frameability): A following game $\mathsf{Game}_{\mathsf{NF}}$ between a $\mathcal{PPT}$ adversary $\mathcal{A}$ and the simulator $\mathcal{S}$ defines the non-frameability of Emularis:

- $\mathcal{S}$ generates the public system parameters pp for $\mathcal{A}$.
- $\mathcal{A}$ gives $(M, \mathcal{L}_{pk}, i)$ to $\mathcal{S}$. $\mathcal{S}$ signs the message M with sk_i obtaining σ, and then send σ to $\mathcal{A}$.
- $\mathcal{A}$ can make adaptive queries to the oracles $\mathcal{CO}, \mathcal{SO}$.
- $\mathcal{A}$ returns a tuple (M^*, σ^*).

We say that $\mathcal{A}$ wins $\mathsf{Game}_{\mathsf{NF}}$ if:

1) σ^* is a valid signature on a message M^* which was not previously queried to the signature oracle $\mathcal{SO}$.
2) Any public key in $\vec{pk}_{\xi} \in \mathcal{L}_{pk}$ was not corrupted by querying to $\mathcal{CO}$ or was not the input of $\mathcal{SO}$.
3) The two signatures σ, σ^* are linked.

Emularis scheme is non-frameable if $\mathsf{Adv}_{\mathcal{A}}^{\mathsf{NF}}$ is negligible where $\mathsf{Adv}_{\mathcal{A}}^{\mathsf{NF}}$ is the advantage of $\mathcal{A}$ winning $\mathsf{Game}_{\mathsf{NF}}$.

D. Security and Privacy Requirements

Our payment scheme should also fulfill a couple of security and privacy requirements in a real-world environment, namely the following.

1) **Identity privacy preservation**. No real identity information of EVs should be leaked, where each EV has many different one-time pseudonyms.
2) **Location privacy preservation**. Emularis in V2G should prevent the disclosure of location privacy, thereby avoiding the security issues it poses.
3) **Amount privacy preservation and correctness**. The amount of balance should be concealed from the public. Any other entity cannot infer the specific amount spent

or received. The receiver is able to verify whether the transferred amount is correct.

4) **Resistant to attacks**. Emularis in V2G should be able to counteract various common attacks, such as the double-spending attack and the slandering attack.

IV. The Proposed Emularis Scheme

A. The Design Details

Based on the aforementioned primitives, the concrete Emularis scheme Π = (`Setup`, `KeyGen`, `Sign`, `Verify`, `Link`) is constructed as described below:

1) Setup(λ): Given a security parameter λ, it selects $\mathbb{G} = \langle G \rangle$ as a additive cyclic group of prime order q, making the underlying DLP is intractable. Let $\mathcal{H}^s(\cdot)$, $\mathcal{H}^p(\cdot)$ below be two secure independent cryptographic hash functions.

$$\mathcal{H}^s(\cdot) : \{0,1\}^* \to \mathbb{Z}_q, \mathcal{H}^p(\cdot) : \{0,1\}^* \to \mathbb{G}$$

Note that $\mathcal{H}_j^s$ is defining as $\mathcal{H}_j^s(x) := \mathcal{H}^s(x \parallel j)$.

2) KeyGen(λ): Given a security parameter λ of the system, this algorithm picks m random numbers $(sk_i^1, \ldots, sk_i^m) \in \mathbb{Z}_q^m$ as private key-vector $\vec{sk}_i$ and computes the public key-vector $\vec{pk}_i = \vec{sk}_i \circ \vec{G} = (sk_i^1 \cdot G, \ldots, sk_i^m \cdot G) = (pk_i^1, \ldots, pk_i^m)$, where $\vec{G} = (G, \ldots, G) \in \mathbb{G}^m$. As a side note, pk_i^j is consistently specified as a one-time address.

3) Sign($\mathcal{L}_{pk}, M, \vec{sk}_\xi$): Given a list of n public key-vectors of length m $\mathcal{L}_{pk} = (\vec{pk}_1, \ldots, \vec{pk}_n)$, a transaction message $M \in \{0,1\}^*$ to be signed, and a private key-vector $\vec{sk}_\xi = (sk_\xi^1, \ldots, sk_\xi^m)$ corresponding to $\vec{pk}_\xi = (pk_\xi^1, \ldots, pk_\xi^m)$, $1 \le \xi \le n$, it generates a Emularis signature as follows.

i) Computes the key images I for all $j \in [m]$, $i \in [n]$

$$H_i^j = \mathcal{H}^p\left(pk_i^j\right)$$

$$I_j = sk_\xi^j H_\xi^j$$

Let $\vec{H} = \left(H_1^1, \ldots, H_1^m, \ldots, H_n^1, \ldots, H_n^m\right)$ and $I = (I_1, \ldots, I_m)$.

ii) Picks random numbers $r_\xi^j, \alpha_\xi^j \in_R \mathbb{Z}_q$ for each $j \in [m]$ and random challenges $c_i \in_R \mathbb{Z}_q$ where $i \in [n] \backslash \{\xi\}$, and then computes for all $j \in [m]$:

$$L_j = r_\xi^j G + \sum_{i=1, i\neq\xi}^{n} c_i pk_i^j$$

$$R_j = \alpha_\xi^j I_j + \sum_{i=1, i\neq\xi}^{n} c_i H_i^j$$

$$c = \mathcal{H}^s\left(M, L_1, \ldots, L_m, R_1, \ldots, R_m\right)$$

The challenge c_ξ is computed as:

$$c_\xi = c - c_{\xi+1} - \cdots - c_n - c_1 - \cdots - c_{\xi-1}$$

Consequently, the equation $c = \sum_{i=1}^n c_i$ holds. Next, the responses s_j, z_j for each $j \in [m]$ are computed as:

$$s_j = r_\xi^j - c_\xi \cdot sk_\xi^j$$

$$z_j = \alpha_\xi^j - c_\xi \cdot \left(sk_\xi^j\right)^{-1}$$

iii) Computes for all $j \in [m]$ to prepare for `NISA` algorithm.

$$b_j = \mathcal{H}_j^s(L_j, R_j, M)$$

$$L = \sum_{j=1}^{m} b_j L_j, R = \sum_{j=1}^{m} b_j R_j$$

$$P = L - \left(\sum_{j=1}^{m} b_j s_j\right) G + R - \left(\sum_{j=1}^{m} b_j z_j I_j\right)$$

where b_j here is used to thwart the Rogue-key attack [36] to some extent.

iv) Let a challenge vector $\vec{a} = (c_1, \ldots, c_n)$. A signer generates a proof π by running the algorithm `NISA.Proof`$(L, \vec{H}, P, c, \vec{a})$.

v) Outputs the final signature $\sigma = (\{s_j\}_{j=1}^m, \{z_j\}_{j=1}^m, \{L_j\}_{j=1}^m, \{R_j\}_{j=1}^m, \{I_j\}_{j=1}^m, \pi)$ on M.

4) Verify($\mathcal{L}_{pk}, M', \sigma'$): When a verifier receives the signature $\sigma' = (\{s_j'\}_{j=1}^m, \{z_j'\}_{j=1}^m, \{L_j'\}_{j=1}^m, \{R_j'\}_{j=1}^m, \{I_j'\}_{j=1}^m, \pi')$ and a public key set $\mathcal{L}_{pk} = (\vec{pk}_1, \ldots, \vec{pk}_n)$ on a transaction message M', it carries out the follow-up actions to verify σ' to be valid.

i) Computes H_i^j for all $j \in [m]$, $i \in [n]$

$$H_i^j = \mathcal{H}^p\left(pk_i^j\right)$$

Let $\vec{H} = (H_1^1, \ldots, H_1^m, \ldots, H_n^1, \ldots, H_n^m)$.

ii) Computes for all $j \in [m]$

$$b_j' = \mathcal{H}_j^s\left(L_j', R_j', M'\right)$$

$$L' = \sum_{j=1}^{m} b_j' L_j', R' = \sum_{j=1}^{m} b_j' R_j'$$

$$P' = L' - \left(\sum_{j=1}^{m} b_j' s_j'\right) G + R' - \left(\sum_{j=1}^{m} b_j' z_j' I_j'\right)$$

iii) The verifier accepts the signature if and only if the algorithm `NISA.Verify`$(L,' \vec{H}, P,' c,' \pi')$ returns 1 where $c' = \mathcal{H}^s(M,' L_1', \ldots, L_m', R_1', \ldots, R_m')$.

Correctness: The following facts demonstrate the correctness of our scheme.

The signature $\sigma = (\{s_j\}_{j=1}^m, \{z_j\}_{j=1}^m, \{L_j\}_{j=1}^m, \{R_j\}_{j=1}^m, \{I_j\}_{j=1}^m, \pi)$ can pass the verification if the equation $P = \vec{a} \cdot \vec{Q}$ holds where the two vectors $\vec{a} = (c_1, \ldots, c_n)$, $\vec{Q} = \{\sum_{j=1}^m b_j \cdot (\sum_{j=1}^m pk_i^j + \sum_{j=1}^m H_i^j)\}_{i=1}^n$. Since $L = \sum_{j=1}^m b_j L_j, R = \sum_{j=1}^m b_j R_j, s_j = r_\xi^j - c_\xi \cdot sk_\xi^j$ and $z_j = \alpha_\xi^j - c_\xi \cdot (sk_\xi^j)^{-1}$, we have:

$$P = L - \left(\sum_{j=1}^{m} b_j s_j\right) G + R - \left(\sum_{j=1}^{m} b_j z_j I_j\right)$$

$$= \sum_{j=1}^{m} b_j \left(r_\xi^j G + \sum_{i=1, i\neq\xi}^{n} c_i pk_i^j\right) - \left(\sum_{j=1}^{m} b_j s_j\right) G$$

$$+ \sum_{j=1}^{m} b_j \left(\alpha_\xi^j I_j + \sum_{i=1, i\neq\xi}^{n} c_i H_i^j\right) - \left(\sum_{j=1}^{m} b_j z_j I_j\right)$$

$$
\begin{aligned}
&= \sum_{j=1}^{m} b_j \left(r_\xi^j G + \sum_{i=1, i\neq\xi}^{n} c_i p k_i^j \right) \\
&\quad - \left(\sum_{j=1}^{m} b_j \left(r_\xi^j - c_\xi \cdot s k_\xi^j \right) \right) G \\
&\quad + \sum_{j=1}^{m} b_j \left(\alpha_\xi^j I_j + \sum_{i=1, i\neq\xi}^{n} c_i H_i^j \right) \\
&\quad - \left(\sum_{j=1}^{m} b_j \left(\alpha_\xi^j - c_\xi \cdot (s k_\xi^j)^{-1} \right) I_j \right) \\
&= \sum_{j=1}^{m} b_j \left(r_\xi^j G + \sum_{i=1, i\neq\xi}^{n} c_i p k_i^j \right) \\
&\quad - \left(\sum_{j=1}^{m} b_j \left(r_\xi^j G - c_\xi \cdot p k_\xi^j \right) \right) \\
&\quad + \sum_{j=1}^{m} b_j \left(\alpha_\xi^j I_j + \sum_{i=1, i\neq\xi}^{n} c_i H_i^j \right) \\
&\quad - \left(\sum_{j=1}^{m} b_j \left(\alpha_\xi^j I_j - c_\xi \cdot H_\xi^j \right) \right) \\
&= \sum_{j=1}^{m} b_j \left(\sum_{i=1}^{n} c_i p k_i^j \right) + \sum_{j=1}^{m} b_j \left(\sum_{i=1}^{n} c_i H_i^j \right) \\
&= \sum_{i=1}^{n} c_i \left(\sum_{j=1}^{m} b_j \cdot \left(\sum_{j=1}^{m} p k_i^j + \sum_{j=1}^{m} H_i^j \right) \right) \\
&= \vec{a} \cdot \vec{Q}
\end{aligned}
$$

Furthermore, we also have:

$$c = \mathcal{H}^s\left(M, L_1, \ldots, L_m, R_1, \ldots, R_m\right) = \sum_{i=1}^{n} c_i = \sum \vec{a}$$

Therefore, we complete the proof of the correctness.

5) Link$(M,' M,'' \sigma,' I,' \sigma,'' I'')$: Given two signatures namely $\sigma'(M') = (\{s_j'\}_{j=1}^m, \{z_j'\}_{j=1}^m, \{L_j'\}_{j=1}^m, \{R_j'\}_{j=1}^m, \{I_j'\}_{j=1}^m, \pi')$ and $\sigma''(M'') = (\{s_j''\}_{j=1}^m, \{z_j''\}_{j=1}^m, \{L_j''\}_{j=1}^m, \{R_j''\}_{j=1}^m, \{I_j''\}_{j=1}^m, \pi'')$, where M' and M'' are some transaction messages, a public verifier after validating the legitimacy of the received signatures, checks if there exists $I_{j'}' = I_{j''}''$. If this is the case, the verifier can draw a conclusion that the two signatures $\sigma,' \sigma''$ are produced by the same signer, which means that a double-spending attack occurs. Therefore, the verifier will discard the second signature.

B. Security Analysis

This part discusses the rigorous security guarantees of Emularis.

Theorem 1 (Unforgeability): Emularis fulfills the unforgeability for any $\mathcal{PPT}$ adversary $\mathcal{A}$ based on the DLP hardness.

Proof: In general, simulator $\mathcal{S}$ is able to solve the DLP by exploiting the adversary $\mathcal{A}$'s ability to successfully forge a ring signature. $\mathsf{Game}_{\mathsf{UNF}}$ between the $\mathcal{PPT}$ adversary $\mathcal{A}$ and simulator $\mathcal{S}$ is as described below:

Setup: Simulator $\mathcal{S}$ generates the public system parameters pp, and selects n users as ring members. Then, $\mathcal{S}$ randomly generates $\vec{sk}_i = (sk_i^1, \ldots, sk_i^m) \in \mathbb{Z}_q^m$ and computes $\vec{pk}_i = \vec{sk}_i \circ \vec{G} = (pk_i^1, \ldots, pk_i^m)(i \in [n])$ as the corresponding public key-vector for each user$_i$. $\mathcal{S}$ sends system parameters pp and a public key set $\mathcal{L}_{pk} = (\vec{pk}_1, \ldots, \vec{pk}_n)$ to $\mathcal{A}$.

Oracle Simulation : $\mathcal{A}$ makes adaptive queries to $\mathcal{RO}$, $\mathcal{CO}$, and $\mathcal{SO}$, but no element in $\mathcal{L}_{pk}$ can be corrupted.

Forge: For the same message M and signer identity, $\mathcal{A}$ successfully forges a signature σ^* that is not queried by $\mathcal{SO}$ to $\mathcal{S}$. Then by the statistical witness-extended emulation of NISA [24], $\mathcal{S}$ is able to extract $(c_1^*, \ldots, c_n^*)$ from σ^*, where $P^* = L^* - (\sum_{j=1}^m b_j^* s_j^*)G + R^* - (\sum_{j=1}^m b_j^* z_j^* I_j^*) = \sum_{j=1}^m b_j^* (\sum_{i=1}^n c_i^* p k_i^j) + \sum_{j=1}^m b_j^* (\sum_{i=1}^n c_i^* H_i^{*j})$. Using the forking lemma [37], $\mathcal{S}$ rewinds to a point that $\mathcal{H}^s(M, L_1^*, \ldots, L_m^*, R_1^*, \ldots, R_m^*)$ is quried to $\mathcal{RO}$ and gets a different $\hat{c}$ instead. $\mathcal{A}$ makes another forgery $\hat{\sigma}$ that contains $(\hat{c}_1, \ldots, \hat{c}_n)$ such that $\hat{c} = \sum_{i=1}^n \hat{c}_i$. The two valid signatures are as described below:

$$\sigma^* = \left(\{s_j^*\}_{j=1}^m, \{z_j^*\}_{j=1}^m, \{L_j^*\}_{j=1}^m, \{R_j^*\}_{j=1}^m, \{I_j^*\}_{j=1}^m, \pi^*\right)$$

$$\hat{\sigma} = \left(\{\hat{s}_j\}_{j=1}^m, \{\hat{z}_j\}_{j=1}^m, \{\hat{L}_j\}_{j=1}^m, \{\hat{R}_j\}_{j=1}^m, \{\hat{I}_j\}_{j=1}^m, \hat{\pi}\right)$$

Both σ^* and $\hat{\sigma}$ are valid signatures, thus we can get $\{L_j^*\}_{j=1}^m = \{\hat{L}_j\}_{j=1}^m$, for all $j \in [m]$:

$$L_j^* = s_j^* G + c_1^* p k_1^j + \cdots + c_n^* p k_n^j = \hat{s}_j G + \hat{c}_1 p k_1^j + \cdots + \hat{c}_n p k_n^j.$$

Then, there exists a index $\xi \in [n]$ where $c_\xi^* \neq \hat{c}_\xi$ (since $\sum_{i=1}^n c_i^* \neq \sum_{i=1}^n \hat{c}_i$) such that $s_j^* G + c_\xi^* p k_\xi^j = \hat{s}_j G + \hat{c}_\xi p k_\xi^j$. Thus we can obtain for the private key $sk_\xi^j = \frac{s_j^* - \hat{s}_j}{\hat{c}_\xi - c_\xi^*}$ which solves the DLP.

Theorem 2 (Anonymity): Emularis is perfectly anonymous as $\mathcal{A}$ with infinite computing power cannot identify the real signer.

Proof: We build a simulator $\mathcal{S}$ to provide perfect anonymity for honst EVs in the random oracle model. EVs' real identity must be revealed if a $\mathcal{PPT}$ adversary $\mathcal{A}$ can win $\mathsf{Game}_{\mathsf{ANO}}$ with non-negligible advantage. As defined in Definition 3, $\mathsf{Game}_{\mathsf{ANO}}$ between a $\mathcal{PPT}$ adversary $\mathcal{A}$ and simulator $\mathcal{S}$ is conducted as described below:

Setup: Simulator $\mathcal{S}$ generates the public system parameters pp, and runs $(\vec{pk}_i, \vec{sk}_i) \leftarrow \mathsf{KeyGen}(\lambda)$ for each $i \in [n]$. Then $\mathcal{S}$ sends system parameters and a public key set $\mathcal{L}_{pk} = (\vec{pk}_1, \ldots, \vec{pk}_n)$ to $\mathcal{A}$.

Oracle Simulation: $\mathcal{A}$ makes adaptive queries to $\mathcal{RO}$, $\mathcal{SO}$.

Challenge. $\mathcal{A}$ gives the tuple $(M^*, \mathcal{L}_{pk}, i_0, i_1)$ to $\mathcal{S}$, where $\vec{pk}_{i_0}, \vec{pk}_{i_1} \in \mathcal{L}_{pk}$. $\mathcal{S}$ picks random numbers $c_1, \ldots, c_n \in_R \mathbb{Z}_q$ and $s_j, z_j \in_R \mathbb{Z}_q$ where $j \in [m]$. $\mathcal{S}$ prepares I_j, H_i^j to compute $L_j = s_j G + \sum_{i=1}^n c_i p k_i^j$ and $R_j = z_j I_j + \sum_{i=1}^n c_i H_i^j$ according to $\mathcal{L}_{pk}$. $\mathcal{S}$ sets $\mathcal{H}^s(M, L_1, \ldots, L_m, R_1, \ldots, R_m) = \sum_{i=1}^n c_i$

in $\mathcal{RO}$. If the hash value is already queried to $\mathcal{RO}$, $\mathcal{S}$ aborts. Otherwise, $\mathcal{S}$ runs NISA algorithm to generate π. Finally, $\mathcal{S}$ picks a random bit $b \in \{0,1\}$, and returns $\sigma = (\{s_j\}_{j=1}^m, \{z_j\}_{j=1}^m, \{L_j\}_{j=1}^m, \{R_j\}_{j=1}^m, \{I_j\}_{j=1}^m, \pi)$ to $\mathcal{A}$.

Guess: Eventually, $\mathcal{A}$ outputs a bit $b^* \in \{0,1\}$ to guess who is the real signer. If $b = b^*$, $\mathcal{A}$ wins $\mathsf{Game}_{\mathsf{ANO}}$.

Since the bit b is not used in signature generation of σ, the probability of $\mathcal{A}$ winning $\mathsf{Game}_{\mathsf{ANO}}$ by random guessing is 1/2. In the above simulation phase, the probability of success (*i.e.* not aborting) in the first query of q_{so} queries to the $\mathcal{SO}$ is $(1 - \frac{q_{ro}}{q})$. The probability of success is $(1 - \frac{q_{ro}}{q})(1 - \frac{q_{ro}+1}{q}) \cdots (1 - \frac{q_{ro}+q_{so}-1}{q}) \geq (1 - \frac{q_{ro}+q_{so}-1}{q})^{q_{so}} \geq 1 - \frac{q_{so}(q_{ro}+q_{so}-1)}{q}$ after q_{ro} queries to $\mathcal{RO}$ and q_{so} queries to $\mathcal{SO}$. Assuming that $q \gg q_{so}(q_{ro} + q_{so} - 1)$, the probability of successful simulation is close to 1, which means that it is almost impossible to expose the knowledge about the secret key to $\mathcal{A}$ during the simulation. In short, if $\mathcal{S}$ does not abort, there exists no $\mathcal{PPT}$ adversary can win $\mathsf{Game}_{\mathsf{ANO}}$ with non-negligible probability more than half.

Theorem 3 (Linkability): Based on the DLP hardness, Emularis fulfills the linkability for any $\mathcal{PPT}$ adversary $\mathcal{A}$.

Proof: Simulator $\mathcal{S}$ is able to solve the DLP by exploiting the adversary $\mathcal{A}$'s ability to successfully break the linkability. The $\mathsf{Game}_{\mathsf{LINK}}$ between the $\mathcal{PPT}$ adversary $\mathcal{A}$ and simulator $\mathcal{S}$ is as described below:

Setup: Simulator $\mathcal{S}$ provides the public system parameters pp to $\mathcal{A}$. Then, $\mathcal{A}$ generates two public key sets $\mathcal{L}_{pk0}$ and $\mathcal{L}_{pk1}$ such that there exists a public key $\hat{pk}$ in both $\vec{pk}^* = (pk_1^*, \ldots, \hat{pk}, \ldots, pk_m^*) \in \mathcal{L}_{pk0}$ and $\vec{pk}' = (pk'_1, \ldots, \hat{pk}, \ldots, pk'_m) \in \mathcal{L}_{pk1}$.

Oracle Simulation: $\mathcal{A}$ makes adaptive queries to $\mathcal{CO}$ and $\mathcal{SO}$, and can corrupt only two certain public keys, namely, $\mathcal{A}$ can only get two private keys $\vec{sk}^*$ and $\vec{sk}'$ corresponding to $\vec{pk}^* \in \mathcal{L}_{pk0}$ and $\vec{pk}' \in \mathcal{L}_{pk1}$, respectively. Thus, $\vec{sk}^*$ and $\vec{sk}'$ contain the same private key $\hat{sk}$.

Challenge: $\mathcal{A}$ outputs two valid signatures $(\mathcal{L}_{pk0}, M_0^*, \sigma_0^*, \{I^*{}_j\}_{j=1}^m)$ and $(\mathcal{L}_{pk1}, M_1^*, \sigma_1^*, \{I'{}_j\}_{j=1}^m)$.

Suppose that $\hat{pk}$ appears as element j^* of $\vec{pk}^*$ and as element j' of $\vec{pk}'$. If $I^*{}_{j^*} = \hat{sk} H^{*j^*}_{\xi} \neq I'_{j'} = \hat{sk} H'^{j'}_{\xi}, \sigma_0^* \neq \sigma_1^*$, it demonstrates that $\mathcal{A}$ takes advantage of $\hat{sk}$ to generate two valid signatures. The unforgeability of Emularis indicates that, under the assumption of the DLP, the probability of $\mathcal{A}$ fabricating a valid signature is negligible. Thus the two signatures include $I^*{}_{j^*} = I'_{j'}$, and Emularis is linkable.

Theorem 4 (Non-frameability): Based on the DLP hardness, Emularis fulfills the non-frameability for any $\mathcal{PPT}$ adversary.

Proof: We suppose that a $\mathcal{PPT}$ adversary $\mathcal{A}$ attempts to forge a signature to link an honest signer's valid signature. We simulate $\mathsf{Game}_{\mathsf{NF}}$ to prove the non-frameability of Emularis.

Setup: Simulator $\mathcal{S}$ generates the public system parameters pp, and runs $(\vec{pk}_i, \vec{sk}_i) \leftarrow \mathsf{KeyGen}(\lambda)$ for each EV_i, $i \in [n]$. Then $\mathcal{A}$ is given the system parameters pp and a public key set $\mathcal{L}_{pk} = (\vec{pk}_1, \ldots, \vec{pk}_n)$ from $\mathcal{S}$. $\mathcal{A}$ sends the tuple $(M, \mathcal{L}_{pk}, \xi)$ to $\mathcal{S}$. $\mathcal{S}$ signs the message M with $\vec{sk}_\xi = (sk_\xi^1, \ldots, sk_\xi^m)$ obtaining σ, and then sends σ to $\mathcal{A}$.

Oracle Simulation: $\mathcal{A}$ makes adaptive queries to $\mathcal{CO}$ and $\mathcal{SO}$, but doesn't query $\mathcal{CO}$ for any public key in $\vec{pk}_\xi = (pk_\xi^1, \ldots, pk_\xi^m)$.

Challenge: $\mathcal{A}$ outputs a forged signature $(M^*, \sigma^* = (\{s_j^*\}_{j=1}^m, \{z_j^*\}_{j=1}^m, \{L_j^*\}_{j=1}^m, \{R_j^*\}_{j=1}^m, \{I_j^*\}_{j=1}^m, \pi^*))$ of the signer ξ.

If σ^* is linked to σ, it implys there exists $I^*_{j^*} = sk_\xi^{*j^*} H_\xi^{j^*} = I_j$. Also, $\mathcal{S}$ uses $\vec{sk}_\xi$ to generate $I_j = sk_\xi^j H_\xi^j$ where $H_\xi^j = \mathcal{H}^p(pk_\xi^j) = H_\xi^{j^*}$. That being said, $\mathcal{A}$ knows one of the private key $sk_\xi^{*j^*} = sk_\xi^j \in \vec{sk}_\xi$. Since $\mathcal{A}$ hasn't queried $\mathcal{CO}$ for any public key in $\vec{pk}_\xi$, $\mathcal{A}$ solves the DLP.

C. Performance Evaluation

1) Implementation: To underline the utility of Emularis, we implement it and analyze its performance from two main indicators, *i.e.*, communication and computation costs, both theoretically and practically. We use the communication cost to be represented by the size of the ring signature. Computation cost is evaluated by the running time of the algorithm.

Experimental Environment: Our python implementation builds the whole scheme. To achieve a trade-off between the security and the practicality of Emularis, we use the Curve25519 elliptic curve which offers 128 bits of security in the `Fastecdsa` library. In the practical evaluation, all the simulation experiments are conducted with Intel(R) Core(TM) i7-10700 CPU of @2.90 GHz and 16.0 GB RAM running 64 bits Ubuntu LTS 18.04 machine.

We make the assumption that there exist n members in the ring signature and m accounts/keys in each member. For concreteness, Table II shows the theoretical complexity. In terms of computation cost, we only consider expensive operations such as exponentiation and hash to group elements. The larger n is, the higher the anonymity level of Emularis is. To make the implementation as simple and comprehensive as possible, we design multiple sets of experiments based on various settings of n and m to evaluate the practical performance from two perspectives.

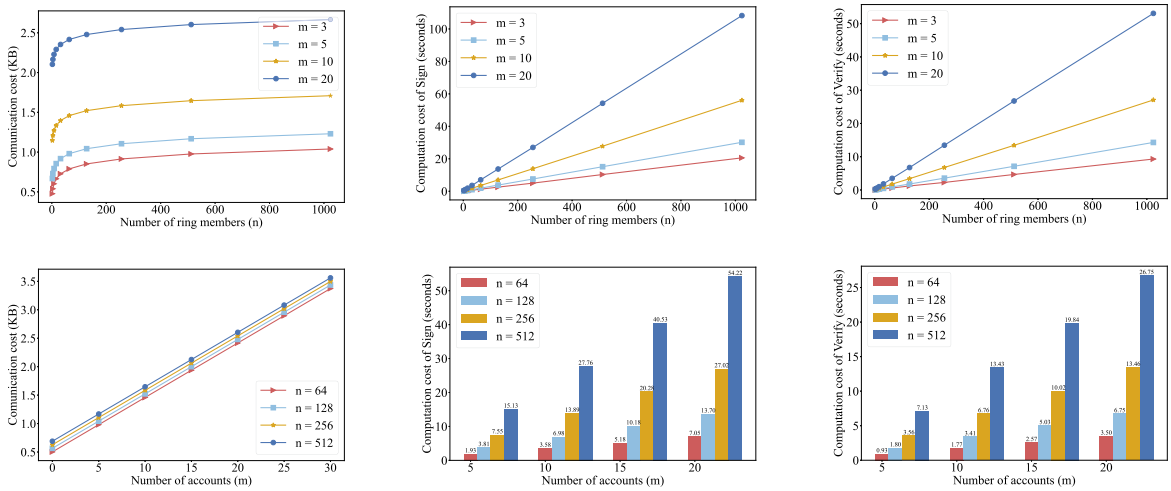
Communication Cost: The theoretical communication complexity of Emularis is $(2m+2)|\mathbb{Z}_q| + (3m + 2logn)|\mathbb{G}|$. It depicts that the communication cost is linear to the number of accounts m while it increases logarithmically with the size of ring members n. The results of practical experiments (see Fig. 3(a)) can bolster the aforementioned conclusion. Assuming that $|\mathbb{Z}_q| = 32$ bytes, $|\mathbb{G}| = 33$ bytes, we intuitively present the communication cost in different sizes of ring members and a different number of accounts, respectively.

Computation Cost: We conduct the matching experiments under four settings in a various number of accounts where m 3; 5; 10; 20 and a various number of ring members where n 64; 128; 256; 512. Fig. 3(b) and (c) demonstrate the practical computation costs of `Sign` and `Verify`, respectively, from which we can deduce that the running time is almost linear to both n and m. As m increases, the running time of our Emularis scheme grows at a faster rate.

TABLE II
COMPARISON WITH DIFFERENT SCHEMES

Scheme	Sign	Verify	Communication
MLSAG [16]	$((4n-1)m)T_{exp}+nT_{hz}$ $+(m(n+1))T_{hg}$	$4mnT_{exp}+T_{hz}+mnT_{hg}$	$(mn+1)\lvert\mathbb{Z}_q\rvert+m\lvert\mathbb{G}\rvert$
CLSAG [39]	$((2+n)m+2+4(n-1))T_{exp}$ $+(m+n)T_{hz}+nT_{hg}$	$((n+2)m+4n)T_{exp}$ $+(m+n)T_{hz}+nT_{hg}$	$(n+1)\lvert\mathbb{Z}_q\rvert+m\lvert\mathbb{G}\rvert$
RingCT 3.0 [23]	$(2log(mn)+6mn+4m+7)T_{exp}$ $+4T_{hz}+T_{hg}$	$(2log(mn)+2mn+5m+13)T_{exp}$ $+4T_{hz}+T_{hg}$	$(m+8)\lvert\mathbb{Z}_q\rvert$ $+(2log(nm)+9)\lvert\mathbb{G}\rvert$
PBT [40]	$((m+1)(n+7logn+1)+logn)T_{exp}$	$((m+1)(n+7logn+3)+logn)T_{exp}$	$(m+1)(3logn+2)\lvert\mathbb{Z}_q\rvert$ $+(m+1)(4logn+1)\lvert\mathbb{G}\rvert$
Ours	$(2m+6n+2logn-2)T_{exp}$ $+(logn+1)T_{hz}+mnT_{hg}$	$(m+2logn+n+3)T_{exp}$ $+(logn+2)T_{hz}+mnT_{hg}$	$(2m+2)\lvert\mathbb{Z}_q\rvert$ $+(3m+2logn)\lvert\mathbb{G}\rvert$

* *"T_{exp}": the time of an exponential operation on in $\mathbb{G}$; "T_{hz}": the time of hash function for $\{0,1\}^* \rightarrow \mathbb{Z}_q$; "$T_{hg}$": the time of hash function for $\{0,1\}^* \rightarrow \mathbb{G}$; "$\lvert\mathbb{Z}_q\rvert$": the size of the element in $\mathbb{Z}_q$, $\lvert\mathbb{G}\rvert$ similarly.*

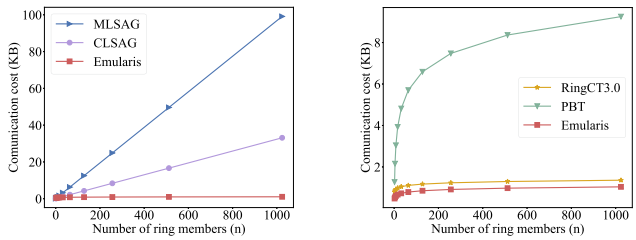


(a) Communication costs (b) Computation costs of Sign (c) Computationn costs of Verify

Fig. 3. Communication costs and computation costs of our Emularis scheme.

2) Comparison: To better illustrate the feasibility and efficiency of Emularis, we carry out a systematic quantitative analysis of it and make a comparison to the existing linear-size ring signature schemes (*e.g.*, the original MLSAG scheme [16], CLSAG scheme [38]) and logarithmic-size ring signature schemes (*e.g.*, RingCT 3.0 [23] and PBT [39] protocol), respectively. Likewise, the comparative experiments take communication and computation costs into account.

Communication Cost: From Table II, the communication complexity of MLSAG, CLSAG, RingCT 3.0, PBT, Emularis is $\mathcal{O}(mn)$, $\mathcal{O}(m+n)$, $\mathcal{O}(lognm)$, $\mathcal{O}(mlogn)$, $\mathcal{O}(m+logn)$ respectively. We observe from Fig. 4(a) that the communication performance in Emularis is better than MLSAG and CLSAG when n is over 4 and 32, respectively. Under the setting of $m=3$, the communication cost in both MLSAG and CLSAG suffers a linear growth with the size of ring members n, while the communication cost in our scheme increases logarithmically with n. Specifically, when the number of n grows larger, Emularis gets more favorable compared to MLSAG and CLSAG. As is clearly shown in Fig. 4(b), compared to the existing ring signature schemes which also produce signatures logarithmic in size, *i.e.*, RingCT 3.0 and PBT, Emularis yields a smaller signature size. Particularly consider a transaction with ring

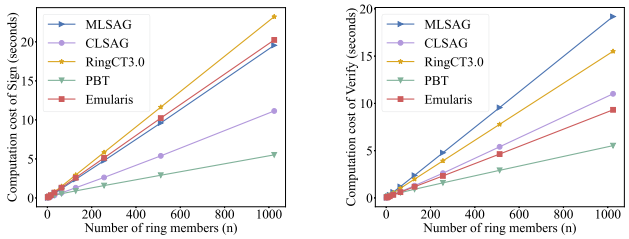


(a) $\mathcal{O}(n)$ size schemes (b) $\mathcal{O}(logn)$ size schemes

Fig. 4. Comparison of communication costs for different numbers of ring members (n) (*s.t.*, number of accounts $m=3$).

members of 1024 (*s.t.*, number of accounts $m=3$), our ring signature size (1.03 KB) is nearly 98.96%, 96.89%, 23.53%, and 88.76% less than the ring signature size of MLSAG (99.12 KB), CLSAG (33.12 KB), RingCT3.0 (1.36 KB), and PBT (9.25 KB), respectively.

In a nutshell, Emularis is much more space-efficient from a practical perspective. Beyond any doubt, since the size grows logarithmically, signers can enhance the anonymity level by expanding the ring size without drastically raising the computation cost. This is a substantial advantage of our Emularis scheme for both theoretical and practical applications.



(a) Computation costs of Sign (b) Computationn costs of Verify

Fig. 5. Comparison of computation costs for different numbers of ring members (n) (*s.t.*, number of accounts $m = 3$).

Computation Cost: As depicted in Fig. 5(a) and (b), extensive simulation results demonstrate that the running time of `Sign` and `Verify` algorithms in MLSAG, CLSAG, RingCT3.0, PBT, and Emularis is all approximately linear to n. The computation cost of `Sign` in Emularis is slightly behind MLSAG, CLSAG, and PBT. This is because of the extra costs brought by the arguments of knowledge algorithm. The computation overhead of `Verify` in Emularis is significantly lower than that of MLSAG, CLSAG, and RingCT 3.0, especially when $n = 1024$, and the running time of `Verify` is greatly reduced by about $\frac{19.1855-9.3181}{19.1855} \approx 51.43\%$, $\frac{11.0130-9.3181}{11.0130} \approx 15.39\%$, and $\frac{15.5103-9.3181}{15.5103} \approx 39.92\%$ respectively. One can find that our Emularis only requires more computation overhead of `Verify` than PBT, but the signature size of Emularis is smaller. As a consequence of the above experimental results, our Emularis scheme, taken as a whole, is more practical and effective than the comparing schemes under consideration.

To summarize, Emularis has better performance than the previous works in terms of communication and computation costs, and thus owns significant advantages for the security-related application in V2G networks.

V. The Anonymous Payment Scheme for V2G

In this section, we first overview how the designed anonymous payment scheme is used in V2G networks, and then give more details about the anonymous payment scheme. Finally, we conduct a simulation to demonstrate its effectiveness.

A. Overview

We design Emularis scheme to achieve anonymous payment in V2G networks. In our anonymous payment scheme, TA is responsible for the system initialization. Each EV needs to register in TA to join the V2G networks for charging services.

Fig. 6 shows the steps to pay for charging. When an EV would like to purchase some electricity from the SG for charging, the EV will follow a specific procedure: An EV first initiates a charging service request including the location of a CS to LAG. When LAG receives the EV charging request, it forwards the request to the SGCC for confirmation. Then, SGCC generates the order message to EV. After receiving the order message from LAG, EV picks another $n-1$ registered EVs and then generates a Emularis signature on the payment transaction message to BS.

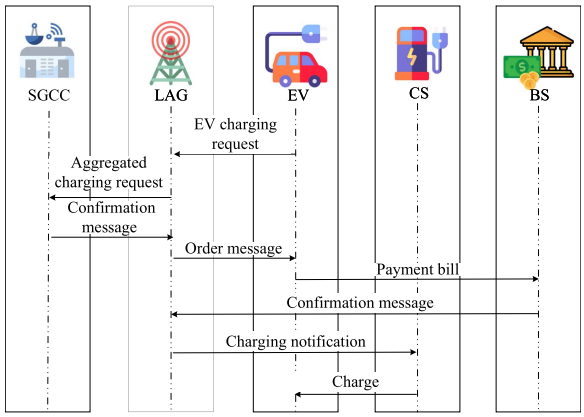


Fig. 6. Steps to pay for charging.

Then, BS sends a confirmation together with the payment transaction message to LAG. This payment transaction information with the Emularis signature is forwarded to LAG. Consequently, LAG decides whether to notify CS to give a charging service by verifying the legitimacy of the signature and transaction.

B. The Basic Procedure of Anonymous Payment

To be exact, the essential idea is to combine Emularis with the RingCT protocol [16]. In this part, we give exact details of our anonymous payment mechanism containing 4 steps, *i.e.*, **Initialize**, **Mint**, **Spend**, and **Verify**. Suppose an EV (named Alice) with m accounts wants to pay for charging in V2G.

1) Initialize: In the initialization process, TA first performs Emularis.`Setup` to obatin the public system parameters pp. Alice runs Emularis.`KeyGen` to get a key pair $(\vec{sk}_\xi = (sk_\xi^1, \ldots, sk_\xi^m), \vec{pk}_\xi = (pk_\xi^1, \ldots, pk_\xi^m))$, and then submits a registration request to TA for the purpose of joining the V2G networks.

2) Mint: After Alice successfully joins the V2G networks, assuming that the electricity price at this time is v_{out}, it picks a random number $rn^j \in \mathbb{Z}_q$ uniformly for each pk_ξ^j, and computes the balance commitment $cm_\xi^j = \mathsf{CMT}(rn^j, v^j) = rn^j \cdot G + v^j \cdot W$, where $j \in [m]$ and v^j is the real amount value in Alice's jth account such that $v_{out} = \sum_j^m v^j$. The balance commitment cm_ξ^j together with pk_ξ^j forms an account (pk_ξ^j, cm_ξ^j).

3) Spend: Alice spontaneously selects another $n-1$ legal EVs to form a ring of size n with each EV containing exactly m different keys. We denote a public key set with n key-vectors by $\mathcal{L}_{pk} = \{\{pk_i^j\}_{j=1}^m\}_{i=1}^n$ and an account set by $ACT = \{\{(pk_i^j, cm_i^j)\}_{j=1}^m\}_{i=1}^n$, respectively. Also, we denote the index of Alice as ξ, *i.e.*, $ACT_\xi = \{(pk_\xi^j, cm_\xi^j)\}_{j=1}^m = \{(sk_\xi^j \cdot G, rn^j \cdot G + v^j \cdot W)\}_{j=1}^m$ with corresponding secret keys $\vec{sk}_\xi = (sk_\xi^1, \ldots, sk_\xi^m)$, and set cm_{out} as the output commitment. The intended output address defaults to the address of BS.

i) Choose $r_{out} \in \mathbb{Z}_q$ at random and compute $cm_{out} = r_{out} \cdot G + v_{out} \cdot W$.

ii) Compute extra public keys $\widehat{pk}_i$ according to ACT and cm_{out} by the following:

$$\widehat{pk}_i = \sum_{j=1}^{m} cm_i^j - cm_{out}$$

$$\widehat{x} = \sum_{j=1}^{m} rn^j - r_{out}$$

Since $\sum_j^m v^j = v_{out}$, we can know that $\widehat{pk}_\xi = \widehat{x} \cdot G$. The public key set $\mathcal{L}_{pk}$ and extra public keys $\widehat{pk}_i$ are arranged into a new set of public keys $\widehat{\mathcal{L}_{pk}}$, in a form of the following matrix.

$$\begin{pmatrix} pk_1^1 & \cdots & pk_\xi^1 & \cdots & pk_n^1 \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ pk_1^m & \cdots & pk_\xi^m & \cdots & pk_n^m \\ \widehat{pk}_1 & \cdots & \widehat{pk}_\xi & \cdots & \widehat{pk}_n \end{pmatrix}$$

iii) Consequently, Alice runs the algorithm Emularis.$\mathtt{Sign}(\widehat{\mathcal{L}_{pk}}, M, \widetilde{sk}_\xi)$ by using $\widehat{\mathcal{L}_{pk}}$ and $\widetilde{sk}_\xi = (sk_\xi^1, \ldots, sk_\xi^m, \widehat{x})$ to output a signature σ over a transaction message M. Then, Alice forwards $(cm_{out}, ACT, M, \sigma)$ to a LAG.

4) Verify: After receiving $(cm_{out}, ACT, M, \sigma)$ from Alice, a LAG verifies whether Alice honestly pays for charging. Above all, the LAG uses $ACT = \{\{(pk_i^j, cm_i^j)\}_{j=1}^m\}_{i=1}^n$ and cm_{out} to compute $\widehat{pk}_i = \sum_{j=1}^m cm_i^j - cm_{out}$. Then, the LAG constructs a public key set $\widehat{\mathcal{L}_{pk}} = (\{\{pk_i^j\}_{j=1}^m\}_{i=1}^n, \{\widehat{pk}_i\}_{i=1}^n)$ to perform Emularis.$\mathtt{Verify}(\widehat{\mathcal{L}_{pk}}, M, \sigma)$. LAG will provide a charging service for Alice if the verification algorithm returns 1.

C. Security Analysis of Anonymous Payment

Based on the above-mentioned theorems in Section IV-B, we further present rigorous analysis according to the security and privacy requirements in Section III-D.

1) **Identity privacy preservation**. Emularis adopts one-time addresses/keys / key for each transaction. To be exact, the one-time address can conceal the real address of an EV. Theorem 2 implies that the $\mathcal{PPT}$ verifier can only get the authorized anonymous public keys rather than the genuine identity due to the anonymity of Emularis.
2) **Location privacy preservation**. No other entity is capable of revealing the identity of EVs. Hence, the location privacy of EVs is preserved.
3) **Amount privacy preservation and correctness**. Our payment scheme can preserve the amount privacy and correctness.

- **Amount privacy preservation.** The amount in our anonymous payment scheme is hidden by using a commitment scheme. According to the hiding property of Pedersen commitment [35], the transaction amount is unknown to others.
- **Amount correctness.** If a malicious EV attempts to spend the balance value $\sum_{j=1}^m v^j$ that is not equal to the acquired electricity bill v_{out}, we know that $\sum_{j=1}^m cm_\xi^j - cm_{out} = (\sum_{j=1}^m rn^j - r_{out}) \cdot G - (\sum_{j=1}^m v^j - v_{out}) \cdot W$. Let $\sum_{j=1}^m v^j - v_{out}$ be y, it is hard to get the discrete logarithm of $\widehat{x} \cdot G - y \cdot W$ in base point G as private key based on the DLP hardness, in turn, the EV cannot generate a valid Emularis signature. That is to say, since the transaction amount satisfies the homomorphic property, a verifier LAG can confirm that the input and output amounts are identical while having no idea of the exact amount.

4) **Resistant to various types of attacks**. We present how our payment mechanism counteracts the following common attacks as follows.

- **Double-spending attack**. A LAG need only verify whether the key images included in Emularis signatures (computed as $I_j = sk_\xi^j \mathcal{H}^p(pk_\xi^j)$) have appeared before in other transactions by running the LINK algorithm. If they have, we can conclude that the signer attempts to spend the balance cm_ξ^j addressed to pk_ξ^j again. In addition, they will uncover the secret index ξ for both transactions where it appears. Consequently, these double-spending signatures will be discarded. Thus, our proposal can satisfy the property of resisting the double-spending attack according to Theorem 3.
- **Slandering attack**. According to Theorem 4, without the knowledge of an honest EV's (denoted as Alice) private key, a malicious $\mathcal{PPT}$ adversary cannot generate the same key image as the signature under Alice's private key. Hence, our scheme could withstand the slandering attack.

Compared to other payment schemes [25], [26] in V2G, our payment scheme is much easier to be expanded into a distributed form to resist the single-point attack, which is a natural fit with the Monero blockchain, without the need of centralized key setup.

D. Simulation

In fact, the scenario we're thinking about is when an EV is moving, it can send a charging request (including the location of a CS) and pay a fee to a nearby LAG to book charging services. If the request verification is passed, the LAG will notify the CS to give charging services for the EV. To show the feasibility of our scheme in a real-life environment, the simulations are performed using NS-2 and VanetMobiSim. The simulated road scenario is in a map of 1.0 $\times$ 1.0 km^2 (see Fig. 7). The EVs are randomly equipped with an average speed of 50 km/h, and the confidence interval of the EV speed is [45, 55] km/h. The communication range of each EV is 300 m. EVs send the transaction message every 100 ms. The maximum broadcast bandwidth is set to 6 Mbps. Considering the actual simulation scenario, we set n to 16 and m to 3, then the packet size is 704 bytes.

For evaluating the simulations, we refer to the average packet delay (APD) and the packet loss ratio (PLR) in [40]. For every number n of EV, we conduct 100 sets of experiments respectively to get the average values of APD and PLR. From Fig. 8, it is noticeable that the APD stays stable at 0.53 s until the number of EVs is greater than 60. A dramatic rise in the PLR takes place

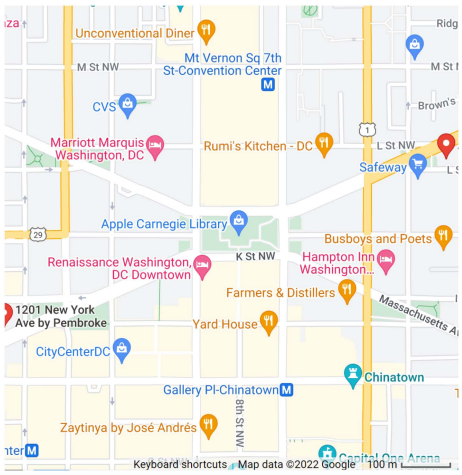


Fig. 7. The map of the simulation scenario.

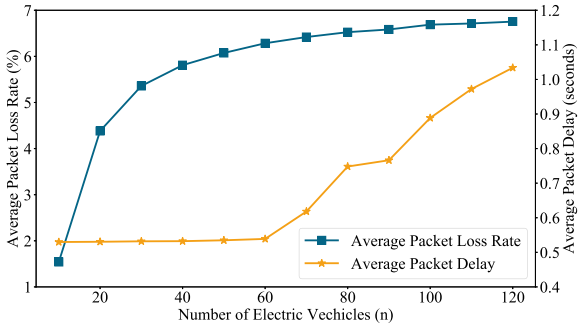


Fig. 8. The packet delay and loss.

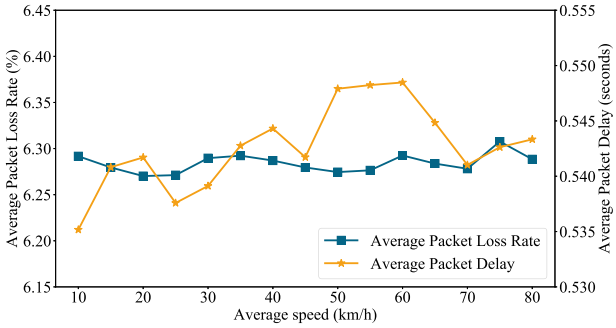


Fig. 9. The impact of vehicle mobility in APD and PLR.

when the number of EVs is less than 35, and then grows slowly as the number of EVs grows. This is due to the limited time for transaction verification. As the number of EVs requesting service rises, LAGs' capacity to process transactions tends to be saturated. Although our scheme has a certain packet loss rate in the actual simulation environment, 6.6% is still acceptable when the number of EVs is 100.

To present how the EVs mobility affects the whole payment activities, we fix the number of EVs at 60 to measure the APD and PLR at different EV average speeds. As can be seen from Fig. 9, the PLR basically stabilizes at 6.27% with the increase of average speed, while the APD fluctuates relatively (but within 13 ms). The PLR is little affected by the EVs mobility. That is, only packets that are out of the communication range will be dropped. Since an EV is charged while parked at the CS, the EVs mobility has little to do with the charging activity.

Through the above analysis, we can demonstrate that our scheme is practical and feasible for use in V2G networks.

VI. Conclusion

We presented a feasible multilayered linkable ring signature, Emularis, based on the sum argument of knowledge. The proposed Emularis achieves a smaller signature size that is logarithmical with the number of ring members. We explained how the proposed signature scheme can be deployed in V2G networks, and the scheme also supports unforgeability, anonymity, linkability, and non-frameability if the underlying crypto-primitive is secure under the random oracle model. We then developed an anonymous payment scheme for V2G using Emularis. The extensive experimental results highlight that Emularis is better suited for V2G due to its comprehensive advantages. We believe with confidence that the improvement of the signature size in our scheme will bring system-wide advancements. For future work, we will study other efficient methods to enhance the performance of our payment scheme.

References

[1] Rinkesh, "What is an electric car?," 2022. [Online]. Available: https://www.conserve-energy-future.com/advantages-and-disadvantages-of-electric-cars.php
[2] F. Kennel, D. Görges, and S. Liu, "Energy management for smart grids with electric vehicles based on hierarchical MPC," *IEEE Trans. Ind. Inform.*, vol. 9, no. 3, pp. 1528–1537, Aug. 2013.
[3] J. Y. Yong, V. K. Ramachandaramurthy, K. M. Tan, and N. Mithulananthan, "A review on the state-of-the-art technologies of electric vehicle, its impacts and prospects," *Renewable Sustain. Energy Rev.*, vol. 49, pp. 365–385, 2015.
[4] V. Hassija, V. Chamola, S. Garg, D. N. G. Krishna, G. Kaddoum, and D. N. K. Jayakody, "A blockchain-based framework for lightweight data sharing and energy trading in V2G network," *IEEE Trans. Veh. Technol.*, vol. 69, no. 6, pp. 5799–5812, Jun. 2020.
[5] Y. Liu, W. Guo, C.-I. Fan, L. Chang, and C. Cheng, "A practical privacy-preserving data aggregation (3PDA) scheme for smart grid," *IEEE Trans. Ind. Inform.*, vol. 15, no. 3, pp. 1767–1774, Mar. 2019.
[6] F. Gao, L. Zhu, M. Shen, K. Sharif, Z. Wan, and K. Ren, "A blockchain-based privacy-preserving payment mechanism for vehicle-to-grid networks," *IEEE Netw.*, vol. 32, no. 6, pp. 184–192, Nov./Dec. 2018.
[7] D. Gabay, K. Akkaya, and M. Cebe, "Privacy-preserving authentication scheme for connected electric vehicles using blockchain and zero knowledge proofs," *IEEE Trans. Veh. Technol.*, vol. 69, no. 6, pp. 5760–5772, Jun. 2020.
[8] R. L. Rivest, A. Shamir, and Y. Tauman, "How to leak a secret," in *Proc. Int. Conf. Theory Appl. Cryptol. Inf. Secur.*, 2001, pp. 552–565.
[9] Y. Zhang, S. Gjessing, H. Liu, H. Ning, L. T. Yang, and M. Guizani, "Securing vehicle-to-grid communications in the smart grid," *IEEE Wireless Commun.*, vol. 20, no. 6, pp. 66–73, Dec. 2013.
[10] H. Liu, H. Ning, Y. Zhang, Q. Xiong, and L. T. Yang, "Role-dependent privacy preservation for secure V2G networks in the smart grid," *IEEE Trans. Inf. Forensics Secur.*, vol. 9, no. 2, pp. 208–220, Feb. 2014.
[11] H. Wang, Q. Wang, D. He, Q. Li, and Z. Liu, "BBARS: Blockchain-based anonymous rewarding scheme for V2G networks," *IEEE Internet Things J.*, vol. 6, no. 2, pp. 3676–3687, Apr. 2019.
[12] Y. Cai, H. Zhang, and Y. Fang, "A conditional privacy protection scheme based on ring signcryption for vehicular ad hoc networks," *IEEE Internet Things J.*, vol. 8, no. 1, pp. 647–656, Jan. 2021.
[13] J. K. Liu, V. K. Wei, and D. S. Wong, "Linkable spontaneous anonymous group signature for ad hoc groups," in *Proc. Australas. Conf. Inf. Secur. Privacy*, 2004, pp. 325–335.
[14] Monero, 2014. [Online]. Available: https://www.getmonero.org/
[15] S. Noether, "CryptoNote v2.0," 2013. [Online]. Available: http://eprint.iacr.org//

[16] S. Noether et al., "Ring confidential transactions," *Ledger*, vol. 1, pp. 1–18, 2016.

[17] Y. Dodis, A. Kiayias, A. Nicolosi, and V. Shoup, "Anonymous identification in ad hoc groups," in *Proc. Int. Conf. Theory Appl. Cryptographic Techn.*, 2004, pp. 609–626.

[18] L. Nguyen, "Accumulators from bilinear pairings and applications," in *Proc. Cryptographers' Track RSA Conf.*, 2005, pp. 275–292.

[19] J. Bootle, A. Cerulli, P. Chaidos, E. Ghadafi, J. Groth, and C. Petit, "Short accountable ring signatures based on DDH," in *Proc. Eur. Symp. Res. Comput. Secur.*, 2015, pp. 243–265.

[20] J. Groth and M. Kohlweiss, "One-out-of-many proofs: Or how to leak a secret and spend a coin," in *Proc. Int. Conf. Theory Appl. Cryptographic Techn.*, 2015, pp. 253–280.

[21] M. F. Esgin, R. Steinfeld, J. K. Liu, and D. Liu, "Lattice-based zero-knowledge proofs: New techniques for shorter and faster constructions and applications," in *Proc. Annu. Int. Cryptol. Conf.*, 2019, pp. 115–146.

[22] R. W. Lai, V. Ronge, T. Ruffing, D. Schröder, S. A. K. Thyagarajan, and J. Wang, "Omniring: Scaling private payments without trusted setup," in *Proc. ACM SIGSAC Conf. Comput. Commun. Secur.*, 2019, pp. 31–48.

[23] T. H. Yuen et al., "RingCT 3.0 for blockchain confidential transaction: Shorter size and stronger security," in *Proc. Int. Conf. Financial Cryptogr. Data Secur.*, 2020, pp. 464–483.

[24] T. H. Yuen, M. F. Esgin, J. K. Liu, M. H. Au, and Z. Ding, "DualRing: Generic construction of ring signatures with efficient instantiations," in *Proc. Annu. Int. Cryptol. Conf.*, 2021, pp. 251–281.

[25] 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.

[26] 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 Secur.*, vol. 10, no. 11, pp. 2340–2351, Nov. 2015.

[27] P. Gope and B. Sikdar, "An efficient privacy-preserving authentication scheme for energy internet-based vehicle-to-grid communication," *IEEE Trans. Smart Grid*, vol. 10, no. 6, pp. 6607–6618, Nov. 2019.

[28] S. Aggarwal, N. Kumar, and P. Gope, "An efficient blockchain-based authentication scheme for energy-trading in V2G networks," *IEEE Trans. Ind. Inform.*, vol. 17, no. 10, pp. 6971–6980, Oct. 2021.

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

[30] J. K. Liu, M. H. Au, W. Susilo, and J. Zhou, "Enhancing location privacy for electric vehicles (at the right time)," in *Proc. Eur. Symp. Res. Comput. Secur.*, 2012, pp. 397–414.

[31] M. H. Au, J. K. Liu, J. Fang, Z. L. Jiang, W. Susilo, and J. Zhou, "A new payment system for enhancing location privacy of electric vehicles," *IEEE Trans. Veh. Technol.*, vol. 63, no. 1, pp. 3–18, Jan. 2014.

[32] N. Saxena, S. Grijalva, V. Chukwuka, and A. V. Vasilakos, "Network security and privacy challenges in smart vehicle-to-grid," *IEEE Wireless Commun.*, vol. 24, no. 4, pp. 88–98, Aug. 2017.

[33] Z. Wan, T. Zhang, W. Liu, M. Wang, and L. Zhu, "Decentralized privacy-preserving fair exchange scheme for V2G based on blockchain," *IEEE Trans. Dependable Secure Comput.*, vol. 19, no. 4, pp. 2442–2456, Jul./Aug. 2022.

[34] B. Bünz, J. Bootle, D. Boneh, A. Poelstra, P. Wuille, and G. Maxwell, "Bulletproofs: Short proofs for confidential transactions and more," in *Proc. IEEE Symp. Secur. Privacy*, 2018, pp. 315–334.

[35] T. P. Pedersen, "Non-interactive and information-theoretic secure verifiable secret sharing," in *Proc. Annu. Int. Cryptol. Conf.*, 1991, pp. 129–140.

[36] G. Maxwell, A. Poelstra, Y. Seurin, and P. Wuille, "Simple schnorr multi-signatures with applications to bitcoin," *Des. Codes Cryptogr.*, vol. 87, no. 9, pp. 2139–2164, 2019.

[37] A. Bagherzandi, J.-H. Cheon, and S. Jarecki, "Multisignatures secure under the discrete logarithm assumption and a generalized forking lemma," in *Proc. 15th ACM Conf. Comput. Commun. Secur.*, 2008, pp. 449–458.

[38] B. Goodell, S. Noether, and A. Blue, "Concise linkable ring signatures and forgery against adversarial keys," *Cryptol. ePrint Arch.*, 2019. [Online]. Available: https://www.getmonero.org/resources/research-lab/

[39] Y. Jia et al., "PBT: A new privacy-preserving payment protocol for blockchain transactions," *IEEE Trans. Dependable Secure Comput.*, vol. 19, no. 1, pp. 647–662, Jan./Feb. 2022.

[40] Y. Liu, L. Wang, and H.-H. Chen, "Message authentication using proxy vehicles in vehicular ad hoc networks," *IEEE Trans. Veh. Technol.*, vol. 64, no. 8, pp. 3697–3710, Aug. 2015.

Yulin Liu received the bachelor's degree from the College of Mathematics and Computer Science, Fuzhou University, Fuzhou, China, in 2021. She is currently working toward the master's degree with the Key Laboratory of Aerospace Information Security and Trusted Computing Ministry of Education, School of Cyber Science and Engineering, Wuhan University, Wuhan, China. Her research interests mainly include information security, applied cryptography, and blockchain.

Debiao He (Member, IEEE) received the Ph.D. degree in applied mathematics from the School of Mathematics and Statistics, Wuhan University, Wuhan, China, in 2009. He is currently a Professor with the School of Cyber Science and Engineering, Wuhan University. He has authored or coauthored more than 100 research papers in refereed international journals and conferences, such as IEEE TRANSACTIONS ON DEPENDABLE AND SECURE COMPUTING, IEEE TRANSACTIONS ON INFORMATION FORENSICS AND SECURITY, and Usenix Security Symposium. His work has been cited more than 10000 times at Google Scholar. His main research interests include cryptography and information security, in particular, cryptographic protocols. He was the recipient of the 2018 IEEE Sysems Journal Best Paper Award and 2019 IET Information Security Best Paper Award. He is on the Editorial Board of several international journals, such as *ACM Distributed Ledger Technologies: Research & Practice*, *Frontiers of Computer Science*, and IEEE TRANSACTIONS ON COMPUTERS.

Zijian Bao received the M.S. degree in computer application technology from the School of Computer Science and Engineering, Northeastern University, Shenyang, China, in 2019. He is currently working toward the Ph.D. degree with the Key Laboratory of Aerospace Information Security and Trusted Computing Ministry of Education, School of Cyber Science and Engineering, Wuhan University, Wuhan, China. His research focuses on cryptographic protocols.

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 research interests include applied cryptography, network security, and cloud computing security.

Muhammad Khurram Khan is currently a Professor of cybersecurity with the Center of Excellence in Information Assurance, King Saud University, Riyadh, Saudi Arabia. He has authored or coauthored more than 400 papers in the journals and conferences of international repute. He is an Inventor of 10 US/PCT patents. He has edited ten books/proceedings published by Springer-Verlag, Taylor& Francis and IEEE. His research interests include cybersecurity, digital authentication, IoT security, biometrics, multimedia security, cloud computing security, cyber policy, and technological innovation management. He is a Fellow of the IET (U.K.), BCS (U.K.), and FTRA (South Korea). He is the Founder and CEO of the 'Global Foundation for Cyber Studies and Research', an independent and non-partisan cybersecurity think-tank, Washington DC, USA. He is the Editor-in-Chief of '*Telecommunication Systems*' published by Springer-Nature with its recent impact factor of 2.314 (JCR 2021). He is also the Editor-in-Chief of the *Cyber Insights Magazine*. He is on the Editorial Board of several journals including, IEEE COMMUNICATIONS SURVEYS & TUTORIALS, *IEEE Communications Magazine*, IEEE INTERNET OF THINGS JOURNAL, IEEE TRANSACTIONS ON CONSUMER ELECTRONICS, *Journal of Network & Computer Applications* (Elsevier), IEEE ACCESS, *IEEE Consumer Electronics Magazine*, *PLOS ONE*, and *Electronic Commerce Research*.

Kim-Kwang Raymond Choo (Senior Member, IEEE) received the Ph.D. degree in information security from the Queensland University of Technology, Brisbane City, QLD, Australia, in 2006. He currently holds the Cloud Technology Endowed Professorship with The University of Texas at San Antonio, TX, USA. He is an ACM Distinguished Speaker and IEEE Computer Society Distinguished Visitor during 2021– 2023. He was the recipient of the IEEE Systems, Man, and Cybernetics Technical Committee on Homeland Security (TCHS) Research and Innovation Award in 2022 and 2019 IEEE Technical Committee on Scalable Computing Award for Excellence in Scalable Computing (Middle Career Researcher). He is also the Founding Co-Editor-in-Chief of the *ACM Distributed Ledger Technologies: Research & Practice,* and the Founding Chair of the IEEE Technology and Engineering Management Society (TEMS)'s Technical Committee on Blockchain and Distributed Ledger Technologies.