A Privacy-preserving Authenticated Key Exchange Protocol for V2G Communications Using SSI

Rohini Poolat Parameswarath, Prosanta Gope, *Senior Member, IEEE*, and Biplab Sikdar, *Senior Member, IEEE*

***Abstract*—The popularity of Electric Vehicles (EVs) has been rising globally in recent years. With the demand for EVs, associated cyber threats are also increasing. Users expose their personal information while charging their EVs, leading to privacy threats. This paper proposes a user-empowered, privacy-aware, authenticated key exchange protocol for Vehicle-to-Grid (V2G) communications. The proposed protocol leverages the concept of Self-Sovereign Identity (SSI) and is based on the concept of Decentralized Identifiers (DID) and Verifiable Credentials (VCs). The use of DIDs empowers users by helping them to have complete control over their identities. The charging station and the user verify that the other party is legitimate by verifying the VC before proceeding with the charging services. Key recovery is another issue we address in this paper. A method to recover lost keys is incorporated into the proposed protocol. The proposed protocol also incorporates an effective user revocation policy. We present formal security proof and informal analysis to show the protocol's robustness against several attacks. We also provide a detailed performance analysis to show that the proposed protocol is efficient.**

***Index Terms*—Accumulator, decentralized identifier (DID), electric vehicle (EV) charging, privacy, user-empowered authentication, user revocation, vehicle-to-grid (V2G), verifiable credential (VC).**

I. Introduction

Vehicle to Grid (V2G) technology refers to technologies that make it possible for electrical energy to flow back and forth between automobiles and the electrical grid. When the vehicle's battery is low, it can be charged from the grid. When the grid needs electricity, electricity can also flow in the opposite direction, from the vehicle to the grid, allowing the vehicles to sell their excess electricity. Electric vehicles (EVs) use bi-directional chargers to support the concept of V2G technology. They take electricity from the grid while charging and return it to the grid from the battery when demand is high [1]. As a result, EVs can also help to lessen the grid's burden during periods of high power demand.

There has been an increased interest in using EVs in recent years. While the registrations of conventional cars fell, electric car registrations increased in 2020 [2]. There are various reasons for this trend. Traditional vehicles run on internal combustion engines that burn hydrocarbons, resulting in air pollution and the greenhouse effect. On the contrary, almost no air pollution is produced by EVs because they use electricity [3]. Hence, authorities across the globe encourage the use of EVs [4] and provide incentives to car users to make the switch to EVs.

Though EVs are better for the environment and pave the path to a sustainable transport system, security and privacy concerns have surfaced with the increased usage of EVs. An attacker may track where users charge their EVs and gather personal details of the users. Tracking the activities of EV users allows the attacker to obtain their footprints. There have been studies on how an attacker can misuse such sensitive information for stalking or physical attacks [5]. Marketing and advertising companies can leverage such information to send unsolicited advertisements.

By allowing only authorized legitimate users to access the EV charging service through authentication helps to address security and privacy challenges. The identity system employed plays an important role in preserving users' privacy. The evolution of identity systems has gone through four phases: centralized identity, federated identity, user-centric identity, and Self-Sovereign Identity (SSI) [6]. A centralized server creates and controls users' identities in centralized identity systems. In such systems, users are locked into the identity-issuing central authority. In other words, centralization gives power to centralized entities. Several federated authorities replace the centralized server of centralized identity systems in federated identity systems. Though it is an improvement over centralized systems, the power and authority of a centralized entity are now distributed among multiple federated authorities. Both centralized and federated identity systems do not give the users complete control over their identities. The next phase, user-centric identity, provided a distinct improvement over the previous two phases. The aim of this phase was to give the right to users to control their identities. However, if a centralized system is used to store identity information, users will not have full control over their identities. The self-sovereign identity followed this. The key driver behind the concept of SSI is that a person's identity must originate from him/her and not from a system that is created to make that person's activities simpler and more streamlined. Also, the users should be able to manage and control their identities. The SSI concept enables people to store their identities on their own devices and present these identities to others at their discretion. SSI achieves user autonomy by leveraging the distributed ledger and the

This work was supported in part by the Ministry of Education, Singapore under grants A-0009040-00-00 and A-0009040-01-00.

R.P. Parameswarath and B. Sikdar are with the Department of Electrical and Computer Engineering, College of Design and Engineering, National University of Singapore, Singapore. (Email: rohini.p@nus.edu.sg, bsikdar@nus.edu.sg)

Prosanta Gope is with the Department of Computer Science, University of Sheffield, United Kingdom. (E-mail: p.gope@sheffield.ac.uk)

cryptography offered by blockchain. The distributed ledger helps with trust in the network and eliminates the need for a central authority, while cryptography ensures security while sharing information among multiple entities in the network.

To tackle the challenges of privacy preservation, remove dependency on a centralized identity system, and establish trust among involved parties, in this paper we leverage the concept of SSI and propose a privacy-aware protocol for V2G. SSI enables individuals to own and control their identities. Decentralized Identifiers (DIDs) [7] and Verifiable Credentials (VC) are basic building blocks of SSI. The proposed protocol is based on DID and VC. The World Wide Web Consortium (W3C) DID Working Group has created a DID standard that allows users to create and manage their identities. They do not have to rely on a central authority for their IDs. The DID ecosystem leverages blockchain technology for the storage of identities in a unified, interoperable, and tamper-proof way. A trusted party can check the legitimacy of the user and sign credentials that other parties can verify digitally before providing services. VCs have been introduced for this purpose. When DID is combined with the concept of VC, any third party can verify the legitimacy of the user. As a result, by combining DID and VC, the proposed protocol ensures that others can verify the identity and credentials of the users through a common root of trust, such as blockchain. Since some users may have to be removed from the list of registered users due to various reasons, the proposed protocol also incorporates an effective user revocation policy using dynamic accumulators. The concepts of Decentralized Identifiers (DID) and Verifiable Credentials (VCs) can be applied to other scenarios such as Vehicle-to-Vehicle (V2V) or truck platooning with suitable modifications, to achieve similar security features.

A. Related Work

Security and privacy preservation in V2G communications have gained significant attention in the literature. The authors of [4] presented a security evaluation of the EV charging framework. The cyber threats faced by different entities and the available defences are also discussed in [4]. Privacy threats in EV charging infrastructure and recommendations to protect users' privacy were presented in [8]. To protect communication between different entities from attacks and preserve users' privacy, many authentication schemes based on cryptographic techniques for V2G communications have been proposed in the literature [9]–[14]. A privacy preservation scheme was proposed in [9] based on the role played by the EV, i.e., customer, storage, or supplier of energy. In each role, privacy concerns are different and addressed separately. An authentication scheme for V2G networks using elliptic curve cryptosystem and bilinear pairing was proposed in [10]. The authors of [1] also proposed a bilinear pairing-based authentication protocol for V2G networks. A data aggregation scheme based on homomorphic encryption was proposed in [11]. An authentication scheme for the EV dynamic charging system was proposed in [12]. In this scheme, EVs purchase anonymous coins from banks that are used for anonymous payment and authentication. A scheme to authenticate EVs when they visit other V2G networks from their home network was proposed in [13]. This scheme based on the bilinear pairing technique helps to preserve the privacy of EVs in the visiting networks. Though the schemes presented in [9]–[13] ensure the privacy of users, most of them use computationally expensive operations. Further, most of these schemes have not considered location forgery attacks by malicious users and malicious charging stations. Abdallah and Shen proposed a computationally less intensive authentication scheme for V2G communications [14]. In their scheme, the power grid ensures the confidentiality and integrity of the exchanged messages. They reduced the computation and communication overhead by reducing the number of exchanged messages during authentication.

On the other hand, several researchers proposed protocols based on Physical Unclonable Functions (PUFs) to secure V2G communications. A privacy-preserving V2G authentication scheme based on PUFs was presented in [15]. PUF-based authentication protocol for the V2G framework was proposed in [16] as well. Hou et al. presented a protocol in [17] that considered integrating a 5G network with the power grid. The protocols in [15]–[17] are lightweight and provide identity privacy. However, they require EVs to be equipped with additional hardware (PUFs).

While charging EVs, a solution to protect users' privacy is to use pseudonyms instead of their real names. Kilari et al. [18] proposed an anonymous authentication framework for V2G communications. In their scheme, the charging stations create pseudonyms for electric vehicles to achieve anonymity. A lightweight authentication protocol for the energy internet (EI) based V2G communication was proposed in [19]. However, Irshad et al. [20] reported that the protocol in [19] is vulnerable to replay and man-in-the-middle attacks and has the limitation of desynchronisation issues. To overcome these limitations, they proposed an improved authentication scheme for the V2G framework in [20]. In the authentication schemes in [19] and [20], the service provider creates the pseudo-identities of the EV users. A blockchain-based authentication scheme for V2G networks that enables anonymous energy trading was proposed in [21]. The scheme in [21] ensures identity privacy and mutual authentication between the participants. In this scheme, the pseudo-identities of EVs are assigned by the utility center, a central authority. The authors of [22] proposed a privacy-preserving authentication protocol for V2G networks based on the nonsupersingular elliptic curve. In other similar schemes, the system master key was generated independently by a trusted third party. If the third party is compromised by an adversary, there is a risk of leakage of the system master key. To address this issue, a key agreement protocol was executed to generate the system master key in [22]. To ensure the identity privacy of EVs, the protocol in [22] uses pseudo-identities for EVs. A third-party authority assigns pseudo-identities to EVs in [22]. A mutual authentication and key exchange scheme for V2G communications based on Elliptic Curve Cryptography (ECC) was proposed in [23]. The protocols in [22] and [23] do not support unlinkability. They are also vulnerable

to user impersonation attacks. The scheme based on the bilinear pairing technique in [24] also used pseudo-identities to preserve the identity of EVs. However, the pseudonyms are updated with the help of a trusted third party.

The electricity suppliers or other trusted authorities assign pseudonyms to users to preserve privacy in [18]–[24]. Though the protocols mentioned above help to achieve some of the security properties in the V2G framework, users depend on a third party for their pseudonyms. None of these protocols considered an important feature, "user-empowerment".

The concept of decentralized identifiers and verifiable credentials has recently gained popularity as a viable privacy preservation method to enable user empowerment. The usage of DID for IoT applications was discussed in [25]. Their research demonstrates the applicability of DIDs even on devices with limited resources. A solution based on DID and VC that allows any interested party, such as an airline operator, to confirm that a person has received a COVID vaccination without compromising his/her privacy was proposed in [26]. Researchers have demonstrated that blockchain, a key enabling technology in DID ecosystem, can help to secure the EV charging infrastructure [27], [28]. The authors of [29] proposed a registration and authentication mechanism for vehicular ad-hoc networks (VANETs) based on blockchain and Decentralized Identifiers. In [30], an authentication scheme for EV charging based on DID and VC was proposed. A privacy-preserving and secure energy trading scheme, based on decentralized identifiers and verifiable credentials, for vehicle-to-vehicle communications was proposed in [31]. Though the schemes presented in [30] and [31] employed DID and VC to empower users, they have not considered a key issue, the usability problem that will lead to the identity loss of the user. The schemes in [30] and [31] have also not addressed user revocation.

B. Motivation

Most of the existing V2G authentication protocols in the literature did not consider the important feature of user empowerment. In the existing works, the pseudo-identities are managed by CSs or other servers where the IDs are created and stored on central servers. Because all sensitive information is stored on a central server in centralized systems, there is a risk of information leakage. There is a privacy threat for users if the server is compromised. There is also a risk of a single point of failure in such centralized systems. To address the above issues, this paper proposes a user-empowered, privacy-preserving, authenticated key exchange protocol for V2G by leveraging the SSI concept. The users do not need to rely on a central issuing authority for their IDs because DIDs are created and controlled by themselves. Thus, we can eliminate the single point of failure issue. The DIDs can be accessed through secure, decentralized platforms such as the blockchain. Further, by using DID, the users are empowered as DID gives them complete control over their identities. Only with their consent and at their discretion can their identities be used, providing a high level of privacy. Since DID enables users to protect their privacy rights, it makes them empowered. Though DIDs and VCs have emerged as promising technologies to enable decentralization and user-empowerment, there are certain challenges to tackle in the DID ecosystem. The DID-based protocols for V2G communications in existing works have not addressed the usability problem and user revocation. The usability problem refers to the scenario where an adversary has access to a user's mobile device for a short duration and tries to delete the private key before being noticed by the user. This will result in the loss of the private key and the identity loss of the user. The schemes presented in [30] and [31] employed DID and VC to build the protocols that help users to create and control their IDs. However, the authors of [30] and [31] have not considered the usability problem in their protocols. The proposed protocol addresses this problem and incorporates a solution for key recovery. The proposed protocol also provides an effective user revocation policy to deregister users if required. The protocols in [30] and [31] have not addressed user revocation. Our protocol also provides protection against unauthorized use of users' mobile devices through two-factor security verification before initiating the authentication process. This biometric verification on the mobile device ensures the user's legitimacy. We also provide an efficient Zero-Knowledge Succinct Non-Interactive Argument of Knowledge (zk-SNARK) proof, which is a privacy-preserving way, to present and verify the VC.

C. Our Contributions

This paper makes the following key contributions:

- **A *new* user-empowerment based privacy-aware authenticated key exchange protocol based on DID and VC** : We propose a new protocol for V2G communications by leveraging the concept of SSI. The proposed protocol enables users to create, manage, and control their identities which empowers them. The decentralized nature of DID eliminates the dependency on a centralized system. The service provider issues VCs to legitimate users. The users present the Zero-Knowledge proof (ZKP) of their VCs to the charging station. The proposed protocol provides several important security properties and establishes a secure session key after successful authentication.
- **Key recovery employing secret sharing technique:** The proposed protocol includes a key recovery mechanism so the user's private key can be stored securely and recovered if lost. We employ Shamir's (k, n) threshold secret sharing scheme [32] for private key recovery.
- **An effective user revocation policy:** Users who no longer require the charging service have to be removed from the list of registered users. The proposed protocol incorporates a user revocation policy that takes into account such users by employing a dynamic universal accumulator. Commuters must prove that they are not on the revocation list to request EV charging.
- **Two-factor security verification:** Users enter their biometrics (e.g., fingerprint) to access mobile devices before initiating the authentication process. This biometric verification on the mobile device ensures the user's

legitimacy. Even if an adversary gets the mobile device, he/she cannot use it for EV charging authentication as the biometric verification will fail.

- **Comprehensive security and performance analysis:** We provide a formal security proof and informal security analysis to demonstrate the proposed scheme's robustness against common attacks. We also provide detailed performance analysis and comparison of the performance of the proposed scheme with other schemes.

The rest of the paper is organized as follows. In Section II, we discuss the preliminaries. In Section III, the system and the adversary models are presented. We present the proposed protocol in Section IV. The formal security proof is presented in Section V, and the informal security analysis is presented in Section VI. We provide performance evaluation and comparison with other schemes in Section VII. The conclusions are drawn in Section VIII.

II. Preliminaries

This section begins with a discussion of the concepts of Decentralized identifiers and Verifiable Credentials. After that, we discuss the cryptographic and mathematical concepts behind the proposed protocol.

Decentralized Identifier: DID is a globally unique identifier that is created by the user. As no centralized authority is required to issue a DID, DID empowers users to have complete control and ownership over their IDs. W3C has developed the standard for DID [7]. A person can have different DIDs to ensure that they are not tracked by correlating their activities. A DID resolves to a DID document. Any decentralized network, such as a blockchain that can resolve a unique key into a unique value can be used to store the DID document. The procedure that maps a DID to the corresponding DID document is called DID resolution. A DID resolver resolves a DID to its DID document. A DID can undergo four operations: 'Read', 'Create', 'Update', and 'Deactivate'. The 'Read' operation is used by the DID resolution function to map a DID into its DID document [7].

The DID document contains information about the DID owner, e.g., the owner's public key. A DID is of the form *did:<DID method>:<method-specific identifier>*. The DID method is a reference to the underlying distributed ledger. The method-specific identifier resolves the DID to the DID document on the ledger. An example of a DID document for the DID **did:<example>:<abcdefghijk>** is shown in Figure 1.

Verifiable Credential: A VC is a set of claims that can be verified using cryptographic techniques such as digital signatures [7]. The concept of VC can be used to verify the claims of one party by another party. The VC ecosystem consists of an *Issuer, a Holder, and a Verifier*. The **Issuer** issues credentials about the VC **Holder** and signs it digitally. Another party, **Verifier**, can verify the statements about the **Holder** [7], [33]. Since cryptographic techniques are used, VCs are tamper-resistant and can be verified digitally by others. An example of a VC is shown in Figure 2. Verifiable Presentation (VP), defined by W3C, enables a user to present the VC to a verifier. VP can be presented using Zero-Knowledge Proof (ZKP).

```
{
 "@context": [
"https://www.xyz.org/did"
]
"id":"did:example:abcdefghijk",
"authentication":[{
"id":did:example:abcdefghijk #keys – 1",
"type": "Ed25519VerificationKey",
"controller": "did:example:abcdefghijk",
"publicKeyMultibase": "4545odsdso0r00re0t0d"
}]
}
```

Fig. 1. An example of a DID document in JSON representation.

```
{
  "@context": [
  "https://www.xyz.org/credentials"
   ]
  "id":"http://ev/credentials/1002",
  "type":["Verifiable Credential", "EVCredential"],
  "issuer": "https://www.xyz.org/issuers/234051",
  "issuanceDate": "2022-01-01:10:10",
  "credentialSubject":{
  "id":did:example:abcdefghijk",
  //claims
  },
  "proof": {
  "type":"ECDSASSignature",
  "proofPurpose":"assertionMethod",
  "created": "2022-01-01:10:10",
  "verificationMethod":"publicKeyoftheIssuerasdfdfjd",
  //digital signature value
  }
}
```

Fig. 2. An example of a Verifiable Credential.

Figure 3 illustrates the workflow of the interaction between DID and VC [7]. The DID subjects create their DIDs. A DID resolves to a DID document as shown in Figure 1 that resides on the blockchain. A trusted *Issuer* issues a signed VC to the user after verifying the user's identity. To receive services from a *Verifier*, the user reveals his/her DID and the VC to the *Verifier*. The *Verifier* verifies the user's VC using the *Issuer*'s public key.

Zero-Knowledge Proof (ZKP): The ZKP enables a person (Prover) to present the knowledge of a value to someone else (Verifier) without revealing any other information. The Zero-Knowledge Succinct Non-Interactive Argument of Knowledge (zk-SNARK) is a ZKP construction where the Prover can prove knowledge of certain information to the Verifier without revealing the actual information and without

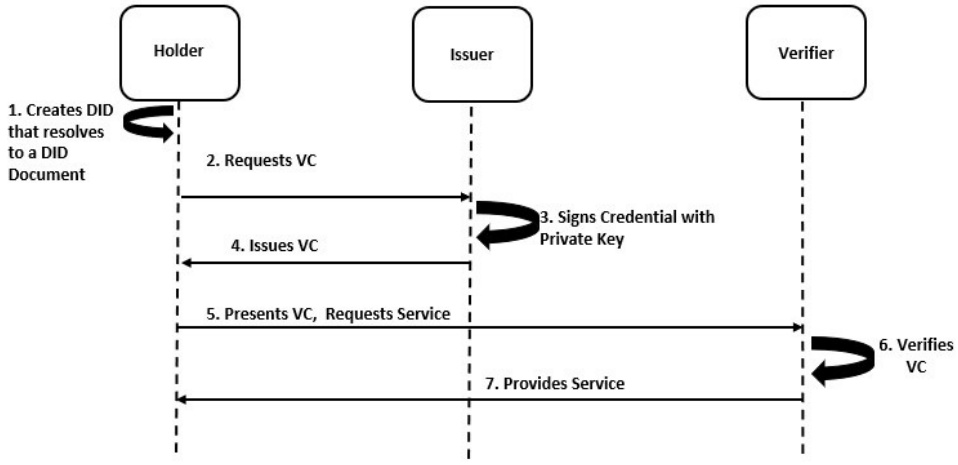

Fig. 3. DID and Verifiable Credential.

any interaction between them. The zk-SNARK system proposed in [34] is one of the most efficient and widely accepted zk-SNARK systems. We use it in the proposed protocol.

A common reference string is shared between the Prover and the Verifier to achieve ZKP [34]–[36], A ZKP consists of the algorithms presented in Table I [34]. In the algorithms mentioned in Table I, a relation generator returns a binary relation R for a security parameter λ. For pairs $(st, w) \in R$, st is the 'statement' and w is the 'witness'. crs is a common reference string and t indicates the simulation trapdoor. There are three properties for ZKPs [34]: completeness, Zero-Knowledge, and soundness.

Completeness: An honest prover can convince a true statement to an honest verifier:

$$Pr[(crs, t) \leftarrow Setup(R); \pi \leftarrow Prove(R, crs, st, w) : Verify(R, crs, st, \pi) = 1] = 1.$$

Zero-Knowledge: The proof does not reveal anything other than the truthfulness of the statement. For all $\lambda \in N$, $(R, z) \leftarrow R(1^\lambda)$, $(st, w) \in R$ and adversary A, we can write:

$$\begin{aligned} &Pr[(crs, t) \leftarrow Setup(R); \pi \leftarrow Prove(R, crs, st, w) : \\ &\qquad A(R, z, crs, t, \pi) = 1] \\ &= Pr[(crs, t) \leftarrow Setup(R); \pi \leftarrow Sim(R, t, st) \\ &\qquad : A(R, z, crs, t, \pi) = 1]. \end{aligned}$$

Soundness: A prover cannot prove a false statement to the verifier:

$$\begin{aligned} &Pr[(R, z) \leftarrow R(1^\lambda); (crs, t) \leftarrow Setup(R); \\ &\qquad (st, \pi) \leftarrow A(R, z, crs) : \\ &\quad st \notin L_R \ \wedge Verify(R, crs, st, \pi) = 1] \approx 0. \end{aligned}$$

Asymmetric Cryptography and ECDSA: The asymmetric cryptographic technique can be used to encrypt or sign data using a pair of keys (public and private). Only the owner has access to the private key. Others have access to the public key. Elliptic-Curve Cryptography (ECC) [37] is a public key cryptography technique. ECC is considered to be secure due to the difficulty of the Elliptic Curve Discrete Logarithm Problem (ECDLP) and Elliptic Curve Decisional Diffie–Hellman problem (ECDDHP) [38], [39] that are defined below.

Elliptic-Curve Discrete Logarithm Problem (ECDLP): Let p be a prime number and $q = p^n$. Let E be an elliptic curve over a finite field F_q. If points $P, Q \in E(F_q)$ are given, ECDLP is the computational problem to find the integer a, if it exists, such that $Q = aP$.

Elliptic-Curve Decisional Diffie-Hellman Problem (ECDDHP): Given $P, aP, bP, Q \in E(F_q)$, ECDDHP is to determine if $Q = abP$.

ECDSA is a popular algorithm used in asymmetric cryptography based on ECC. In the proposed protocol, we use ECDSA. Cryptosystems based on elliptic curves use small key sizes. They have low memory usage and need less processor resources. Due to these properties, they are ideal even for resource-constrained devices. ECDSA signature generation and verification algorithms are shown in Algorithms 1 and 2, respectively [40]. The private key is a random integer in the range $0, \ldots, n-1$. The public key is calculated as $pubK = priK \times G$ where G is the generator point. In Algorithm 1, a message m is signed with $priK$ to get the signature (r, s). In Algorithm 2, the message m, the signature, and the public key are the inputs. The signature is accepted or rejected based on whether it is valid or not.

Algorithm 1 ECDSA Signature Generation, $Sign_{ECDSA}(m, priK)$ -> (r, s)

Calculate $h = hash(m)$
Generate a random number k between 1 and $n-1$
Calculate $R = k \times G$ and find its x-coordinate $r = R.x$
The signature $s = k^{-1} \times (h + r \times priK)(\mod n)$
Signature is (r, s)

Key Recovery and Shamir's Secret Sharing: The DID ecosystem is built on public key infrastructure where users need to manage their private keys. Users should have a secure backup of their private keys. Key possession and

TABLE I
ZKP ALGORITHMS

Algorithm	Description
$(crs, t) \leftarrow Setup(R)$	For the relation R, crs and t are generated.
$\pi \leftarrow Prove(R, crs, st, w)$	This algorithm takes crs and $(\text{st}, w) \in R$ as inputs and returns an output π.
$0/1 \leftarrow Verify(R, crs, st, \pi)$	crs, st, and π are the inputs for this algorithm, and 0 (reject) or 1 (accept), is the output.
$\pi \leftarrow Sim(R, t, st)$	The simulator takes t and st as inputs and returns π.

Algorithm 2 ECDSA Signature Verification, $Verify_{ECDSA}(m, pubK, Signature)$-> $(Accept, Reject)$

Calculate $h = hash(m)$
Calculate $s^{'} = s^{-1}(\mod n)$
Calculate $R^{'} = (h \times s^{'}) \times G + (r \times s^{'}) \times pubK$ and its x-coordinate $r^{'} = R^{'} \times x$
if $r^{'} = r$ **then**
  $Accept$
else
  $Reject$
end if

recovery of lost keys are crucial in maintaining trust [41]. Key recovery refers to methods for securely backing up private keys so that they can be recovered if the private key is lost in events such as the device containing the private key is damaged. We use Shamir's threshold scheme [32] for private key recovery. In this scheme, the key is encoded into a polynomial. After that, it is divided into components. Using polynomial interpolation, the key can be computed with a threshold value of the components.

In this method, a key D is split into n components $D_1, D_2, \cdots, D_n$ such that:

(1) D can be computed from any k (threshold value) or more pieces and

(2) D cannot be calculated when there are only $k-1$ or fewer pieces.

To divide D into k shares, the holder of the key selects a polynomial of degree $k-1$ as:

$$f(x) = D + a_1(x) + \cdots + a_{k-1}(x)^{k-1}. \tag{1}$$

In (1), $a_1, a_2, \cdots, a_{k-1}$ are random polynomial coefficients. After that, the n values are evaluated as $D_1 = f(1)$, $D_2 = f(2)$, $\cdots$, $D_n = f(n)$. With any subset of k pairs from the set of these n values of $\{(i, f(i))\}$, the key can be calculated as

$$D = \Sigma_{j=1}^{k} f(i_j) \Pi_{j \neq m} \frac{i_j}{i_j - i_m}. \tag{2}$$

By knowing $k-1$ or fewer of these values, D cannot be calculated. This method of secret sharing can be employed for key recovery.

Dynamic Accumulator: In the accumulator scheme, many values are condensed into a single short value. A witness exists for values included in the accumulator [42]. A dynamic accumulator allows adding or removing values dynamically [42]. A dynamic universal accumulator where a witness exists for non-members was proposed in [43]. The proposed protocol incorporates a revocation policy using dynamic universal accumulators [42], [43] to remove the users who leave the EV charging service. The list of revoked users is kept in the accumulator. The users must provide a non-membership witness that they are not in the accumulator before requesting EV charging. The features of the accumulator used in the proposed protocol are listed below [43]:

Generating an Accumulator: A secret key k_{Acc} is generated. The accumulator generation function $G_{Acc}()$ takes k_{Acc} and the revocation list L as inputs and returns an accumulator U. This function can be written as $U \leftarrow G_{Acc}(k_{Acc}, L)$.

Updating an Accumulator: To add a new value v_{new} to the accumulator, this function $Upd_{Acc}()$ takes the current accumulator U, k_{Acc}, and v_{new} as the inputs and outputs the updated accumulator U_{new}. This function can be written as $U_{new} \leftarrow Upd_{Acc}(U, k_{Acc}, v_{new})$.

Generation of a Non-membership Witness: The non-membership witness generation function G_w takes U, k_{Acc}, the revocation list L, and a value a that is not in L and produces a non-membership witness w for a. This function can be written as $w \leftarrow G_w(U, k_{Acc}, L, a)$.

Verification of a Non-membership Witness : The non-membership witness verification function $V_w()$ takes U, a witness w for a that is not in L, and a. If a is not in L, V_w outputs 1. If a is in L, V_w outputs 0. $V_w()$ can be written as $0/1 \leftarrow V_w(U, w, a)$.

III. SYSTEM AND ADVERSARY MODEL

A. System Model

The V2G system model considered in this paper is illustrated in Figure 4. The service provider (SP), the charging station (CS), and the users are the participants in this model. The SP is responsible for producing power, distributing it to CSs, and managing data. Electricity is generated from different sources such as wind farms, solar farms, and hydroelectric plants. Then, it is transmitted and distributed to the CSs. The charging rate at a CS depends on its location. The users use their mobile devices (MDs) to connect to the Internet. All the participants in the model communicate with each other through the Internet. Each participant owns DIDs. The corresponding DID documents are stored on the blockchain.

B. Adversary Model

During authentication, the participants exchange messages over an insecure channel, the Internet, over which attackers may launch multiple attacks. According to the Dolev-Yao threat model, an adversary can listen, modify, or delete the

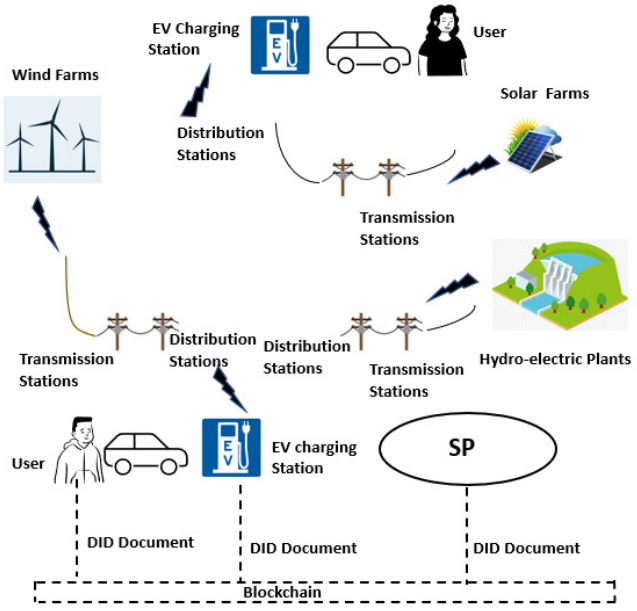

Fig. 4. System model.

messages sent between different parties. Hence, the following threats exist against the system model:

Data Modification Threat: Since the attacker is able to eavesdrop, edit or delete the messages exchanged, there is a threat of data modification. A legitimate user may be denied access due to such data modification.

Unauthorized Access Threat: The attacker can capture legitimate messages and replay them later to get authenticated as a legitimate user. Also, an attacker may impersonate a registered user to charge the vehicle.

Privacy Threat: An adversary may collect information regarding the user's presence in a location at a particular time, his/her daily routines and trajectory, and charging pattern, from the data related to the user's EV charging.

Stealing of Mobile Device Threat: The adversary has the capability to steal the mobile device of a legitimate user. Then, the adversary may use it to initiate EV authentication while impersonating the user.

Loss of Private Key Threat: We also consider the scenario of 'Usability Problem' where the adversary has access to the mobile device of a legitimate user for a couple of minutes, and tries to delete the private key before being noticed by the owner of the device. This will result in the loss of the private key and the identity loss of the user.

Location Forgery Threat: The SP can be trusted in the proposed model, but the other two parties may not be trustworthy. A user who cannot be trusted may present the incorrect location area identifier (L) of the CS to pay less than the actual rates for EV charging. A dishonest CS may also present an incorrect L to obtain higher EV charging rates from users. As a result, location forgery threat also exists.

IV. Proposed Authentication Protocol

The privacy-preserving authenticated key exchange protocol for V2G is presented in this section. The participants in this protocol are the SP, the user, and the CS. The SP issues VCs to users upon registration, and the CS verifies the VC of the user to confirm the user's legitimacy during the authentication process. The phases in the proposed protocol are *system setup and registration*, *authentication*, *key recovery*, and *user revocation*. The system setup and registration phase is only performed once. Before charging their vehicles, registered users are required to go through the authentication process. The key recovery phase is executed if the user wants to retrieve a lost private key. The user revocation phase is executed to revoke a registered user. The proposed protocol is built on the DID and VC standard given in [7]. The high-level workflow of the EV charging protocol can be explained based on Figure 3. The user creates his/her DID that resolves to a DID document. Then, the user presents a signed digital identity to the SP for verification. If the digital identity verification is successful, the SP generates a credential. Then, the SP signs the credential with its private key and issues the signed VC to the user. The user generates the ZKP of the VC. Then, the user sends the charging request and the ZKP of the VC to the CS. The charging station verifies the received ZKP. If the verification is successful, the CS grants the request to the user. The notations used in the proposed authentication protocol are presented in Table II.

TABLE II
Notations

Symbol	Description
G	Generator point
$a_1, a_2, .., a_{k-1}$	Polynomial coefficients for key recovery
DID_x	DID of x
$PDID_i$	Pseudo-identity of $User_i$
K_x^{pr}	Private key of x
K_x^{pu}	Public key of x
MD_i	Mobile device of $User_i$
r	Nonce generated by the SP
K_i	Secret key between SP and $User_i$
K_{CSj}	Secret between SP and CS_j
$[VC_i]_{K_{SP}^{pr}}$	VC of $User_i$
SK	Session key
$\|$	Concatenation operation
$\oplus$	XOR operation
$h(X)$	Hash of X

A. Assumptions

We assume that a secure channel is used for communication between the parties in the registration phase. We also assume that the SP can be trusted. Another assumption is that the CS and the SP do not collude to gather details of users. The users must be authenticated before charging their EVs each time, even if they have previously charged from the same charging station. The proposed protocol is also applicable in shared vehicle environments. In this scenario, each user must be a valid registered user and must use their own VCs to get authenticated.

B. System Setup and Registration Phase

1) System Setup: The system setup comprises the following steps:

TABLE III
EV REGISTRATION PHASE

User		SP
Generate: DID_i, $PDID_i$ Generate: K_i^{pr}, K_i^{pu} pair M_{REV1}:		
$\{Reg_{req}, [VC_{DigIDi}]_{K_{Gov}^{pr}}, DID_i, PDID_i\}$	$\xrightarrow{M_{REV1}}$	$Verify_{ECDSA}$: $[VC_{DigIDi}]_{K_{Gov}^{pr}}$ Generate: nonce n, credential $cred$ $hV = h(cred \parallel K_i^{pu} \parallel n)$ Sign hV with K_{SP}^{pr} i.e., $[VC_i]_{K_{SP}^{pr}} = Sign_{ECDSA}(hV)$ $w_i = G_w(U, k_{Acc}, L, PDID_i)$ Generate: K_i Store: DID_i, $PDID_i$, n, w_i, K_i
Compute and Store: $\delta_i = h(b_i \parallel p_i)$ Compute: $K_i^* = K_i \oplus h(b_i \parallel p_i)$ Store: K_i^*, $[VC_i]_{K_{SP}^{pr}}$, n, cred, w_i	$\xleftarrow{M_{REV2}}$	$M_{REV2} : \{K_i, [VC_i]_{K_{SP}^{pr}}, n, cred, w_i\}$

Step 1: The DID document corresponding to the SP's DID, DID_{SP}, is stored on the blockchain. The SP generates its private key K_{SP}^{pr} as a random integer. The public key is generated through ECDSA key generation function. Let G be the generator point. The public key K_{SP}^{pu} is a point on the elliptic curve, calculated by the Elliptic Curve (EC) point multiplication as $K_{SP}^{pu} = K_{SP}^{pr} \times G$. K_{SP}^{pu} is stored in the SP's DID document.

Step 2: $User_i$ generates a DID (DID_i) and stores the corresponding DID document on the blockchain. Then, $User_i$ generates a pair of private (K_i^{pr}) and public (K_i^{pu}) keys using the ECDSA key generation function similar to the key generation mentioned in Step 1. Let the user's mobile device be denoted as MD_i. The user stores K_i^{pr} in the digital wallet on MD_i. K_i^{pu} is stored in $User_i$'s DID document.

Step 3: The charging station, CS_j also generates its DID (DID_{CSj}) and stores its public key immutably in its DID document on blockchain.

Step 4: The SP generates the accumulator. First, the SP generates a revocation list L and a secret key k_{Acc}. Initially, L does not have any elements. As mentioned in Section II, the SP generates an accumulator U from k_{Acc} and L using the G_{Acc} function.

2) Registration: In this phase, the registration of the electric vehicles and the charging stations with the SP takes place.

EV Registration: The registration of the EV consists of the following steps:

Step 1: $User_i$ generates a pseudo-identity $PDID_i$. The user holds $[VC_{DigIDi}]_{K_{Gov}^{pr}}$, the VC of his/her digital identity issued by a trusted party (e.g., a government agency). Then, the user generates a message M_{REV1} with a registration request, DID_i, the pseudo-identity $PDID_i$, and $[VC_{DigIDi}]_{K_{Gov}^{pr}}$. After that, $User_i$ sends M_{REV1} to the SP.

Step 2: The SP verifies the user's digital identity using the ECDSA's signature verification algorithm as mentioned in Algorithm 2. After that, the SP generates a nonce n and a credential $cred$. Then, the credential, K_i^{pu}, and n are concatenated and its hash value hV is computed. Then, the SP signs $cred$ with its private key using the ECDSA's signature generation algorithm as mentioned in Algorithm 1. For that, the SP generates a random number k between 1 and $m-1$. The random point R and its x-coordinate r are calculated as:

$$R = k \times G; r = R.x. \tag{3}$$

Then, the SP calculates the signature as:

$$k^{-1} \times (hV + r \times K_{SP}^{pr}) (\text{mod } m). \tag{4}$$

The verifiable credential $[VC_i]_{K_{SP}^{pr}}$ of $User_i$ given in (4) indicates that $User_i$ is a valid registered user. It can be verified using the corresponding public key K_{SP}^{pu} of the SP. The SP also generates a non-membership witness w_i for $User_i$ by calling $G_w()$ with inputs U, k_{Acc}, L, and $PDID_i$ as mentioned in Section II.

The SP generates a key K_i and then stores DID_i, $PDID_i$, w_i, and K_i for future communication with $User_i$. After that, the SP generates a message M_{REV2} with K_i, $cred$, n, w_i, and the signed VC and sends it to the user.

Step 3: The user, $User_i$, receives M_{REV2} from the SP. Then, $User_i$ enters his/her biometrics (e.g., fingerprint) b_i and password p_i. This stored input will be used during the authentication phase to ensure the user's legitimacy. Then, $User_i$ computes $K_i^* = K_i \oplus h(b_i \parallel p_i)$. After that, $User_i$ stores K_i^*, $cred$, n, w_i, and $[VC_i]_{K_{SP}^{pr}}$ in the digital wallet on MD_i.

Table III depicts the EV registration steps in detail.

CS Registration: The following are the steps involved in CS registration:

Step 1: Similar to EV registration, CS_j registers with the SP and receives a verifiable credential VC_{CSj} signed by the SP with K_{SP}^{pr}.

Step 2: The SP produces a key K_{CSj} and sends it to CS_j.

Step 3: The CS stores K_{CSj} to use during the authentication phase.

C. Authentication Phase

The user $User_i$ must be authenticated by CS_j to charge his/her vehicle at CS_j. $User_i$ also must authenticate CS_j.

Step 1: $User_i$ enters his/her biometrics b_i and password p_i into MD_i. Then, MD_i computes $\delta_i' = h(b_i \parallel p_i)$ and compares δ_i' with δ_i to verify $User_i$'s legitimacy. This biometric verification on the mobile device is to prevent the unauthorized use of the mobile device by an adversary. After successful biometric verification, MD_i calculates $K_i = K_i^* \oplus h(b_i \parallel p_i)$. Subsequently, $User_i$ generates M_{A1} with $PDID_i$, request for EV charging, and request for the CS's VC. Then, M_{A1} is sent to CS_j. CS_j composes and sends a message M_{A2} to $User_i$ with DID_{CSj}, $[VC_{CSj}]_{K_{SP}^{pr}}$, and a request of proof of $User_i$'s VC.

Step 2: $User_i$ receives M_{A2} from CS_j. Then, $User_i$ verifies CS_j's VC by using the ECDSA's signature verification algorithm with K_{SP}^{pu} as mentioned in Algorithm 2. After that, credential $cred$, K_i^{pu}, and the nonce n are concatenated. Then, the resultant value's hash is computed. Then, $User_i$ generates a nonce N_i and computes $EL_i = L_i \oplus h(K_i \parallel N_i)$ where L_i is the location area identifier of $User_i$. After that, $User_i$ computes $V_1 = h(PDID_i \parallel N_i \parallel K_i \parallel EL_i)$. Next, $User_i$ produces a ZKP of the VC using the *Prove* algorithm with an output π as mentioned in Table I. This is the ZKP of the VC that $User_i$ presents to CS_j to prove in Zero-Knowledge that he/she holds a valid VC. Then, the user composes $M_{A3} : \{hV, \pi, N_i, EL_i, V_1, w_i\}$ and sends it to CS_j.

When CS_j receives M_{A3}, it verifies π using the *Verify* algorithm as mentioned in Table I. Thus, $User_i$ presents the credential to CS_j using ZKP. The CS can only know that $User_i$ holds a valid credential. It cannot learn anything more than that. Then, CS_j verifies the non-membership witness w_i of the user by calling the V_w function as mentioned in Section II to confirm that the user is not revoked. Subsequently, CS_j generates a nonce N_{CSj} and sends a message to the SP with its location area identifier L_{CSj}. For that, CS_j computes $V_2 = h(DID_{CSj} \parallel N_{CSj} \parallel K_{CSj} \parallel L_{CSj})$. After that, CS_j prepares a message $M_{A4} : \{M_{A3}, DID_{CSj}, N_{CSj}, L_{CSj}, V_2\}$. Then, CS_j sends M_{A4} to the SP.

Step 3: The SP checks $PDID_i$, V_1, and V_2. After that, the location identifier received from $User_i$, L_i, and from CS_j, L_{CSj}, are compared by the SP. The SP wants to verify that L_i and L_{CSj} are the same before proceeding further. This step prevents the location forgery attack. Then, the SP generates a session key SK and a nonce n^{new}. Subsequently, the credential, K_i^{pu}, and n^{new} are concatenated and its hash value, hV^{new}, is calculated. After that, the SP generates a new VC for $User_i$ based on the new hash value hV^{new} that the user can use in the next charging authentication process. The user cannot be tracked because the VC for each charging event is unique. The SP saves $User_i$'s PDID for the following round as $PDID_i^{new} = h(PDID_i \parallel K_i)$. Then, the SP generates a non-membership witness w_i^{new} for $PDID_i^{new}$. The session key to send to $User_i$ is generated as $SK_i = h(PDID_i \parallel N_i \parallel K_i) \oplus SK$. Similarly, the session

TABLE IV
AUTHENTICATION PHASE

User
Compute: $\delta_i^* = h(b_i \parallel p_i)$
Check: $\delta_i = \delta_i^*$
Compute: $K_i = K_i^* \oplus h(b_i \parallel p_i)$
MA_1: $\{PDID_i, Usage_{Req}, VC_{Req}\}$
Send MA_1 to CS
CS
MA_2: $\{DID_{CSj}, [VC_{CSj}]_{K_{SP}^{pr}}, Proof_{Req}\}$
Send MA_2 to User
User
$Verify_{ECDSA} : [VC_{CSj}]_{K_{SP}^{pr}}$
Calculate: $hV = h(cred \parallel K_i^{pu} \parallel n)$
Store: hV
Generate: N_i
$EL_i = L_i \oplus h(K_i \parallel N_i)$
$V_1 = h(PDID_i \parallel N_i \parallel K_i \parallel EL_i)$
$\pi = Prove([VC_i]_{K_{SP}^{pr}})$
MA_3: $\{hV, \pi, N_i, EL_i, V_1, w_i\}$
Send MA_3 to CS
CS
Verify: π
$V_w(U, w_i, PDID_i) =?1$
Generate: N_{CSj}
$V_2 = h(DID_{CSj} \parallel N_{CSj} \parallel K_{CSj} \parallel L_{CSj})$
MA_4: { MA_3, DID_{CSj}, N_{CSj}, L_{CSj}, V_2 }
Send MA_4 to SP
SP
Check: $PDID_i$, V_1, V_2
Compare: L_i with L_{CSj}
Generate: SK, n^{new}
Calculate: $hV^{new} = h(cred \parallel K_i^{pu} \parallel n^{new})$
$[VC_i]_{K_{SP}^{pr}}^{new} = Sign_{ECDSA}(hV^{new})$
$PDID_i^{new} = h(PDID_i \parallel K_i)$
$w_i^{new} = G_w(U, k_{Acc}, L, PDID_i^{new})$
$SK_i = h(PDID_i \parallel N_i \parallel K_i) \oplus SK$
$SK_{CSj} = h(DID_{CSj} \parallel N_{CSj} \parallel K_{CSj}) \oplus SK$
$V_3 = h(SK_{CSj} \parallel N_{CSj} \parallel K_{CSj})$
$V_4 = h(SK_i \parallel N_i \parallel K_i)$
$n^{*new} = K_i \oplus n^{new}$
$[VC_i]_{K_{SP}^{pr}}^{*new} = K_i \oplus [VC_i]_{K_{SP}^{pr}}^{new}$
Store : $PDID_i^{new}, n^{new}, w_i^{new}$
$M_{A5i} : (SK_i, V_4, n^{*new}, w_i^{new}, [VC_i]_{K_{SP}^{pr}}^{*new})_{Enc_{K_i^{pu}}}$
$M_{A5} : \{M_{A5i} \parallel (SK_{CSj}, V_3)\}$
Send MA_{A5} to CS
CS
Compute and Verify: V_3
$SK = h(DID_{CSj} \parallel N_{CSj} \parallel K_{CSj}) \oplus SK_{CSj}$
$M_{A6} = M_{A5i}$
Send MA_{A6} to User
User
Compute and Verify: V_4
$SK = h(PDID_i \parallel N_i \parallel K_i) \oplus SK_i$
$PDID_i^{new} = h(PDID_i \parallel K_i)$
$n^{new} = K_i \oplus n^{*new}$
$[VC_i]_{K_{SP}^{pr}}^{new} = K_i \oplus [VC_i]_{K_{SP}^{pr}}^{*new}$

key to send to CS_j is generated as $SK_{CSj} = h(DID_{CSj} \parallel N_{CSj} \parallel K_{CSj}) \oplus SK$. After that, the SP computes a response $V_3 = h(SK_{CSj} \parallel N_{CSj} \parallel K_{CSj})$ for CS_j and a response $V_4 = h(SK_i \parallel N_i \parallel K_i)$ for $User_i$. The new nonce and the VC are XORed with K_i before sending them to $User_i$. After that, the SP generates a message M_{A5i} by encrypting $\{(SK_i, V_4, n^{*new}, w_i^{new}, [VC_i]_{K_{SP}^{pr}}^{*new})\}$ with K_i^{pu}. Then, the SP generates a message $M_{A5} : \{M_{A5i} \parallel (SK_{CSj}, V_3)\}$. Then, the SP sends M_{A5} to the CS.

Step 4: When CS_j receives M_{A5}, it verifies V_3. Then, CS_j calculates the SK as $SK = h(DID_{CSj} \parallel N_{CSj} \parallel K_{CSj}) \oplus SK_{CSj}$ and sends $M_{A6} : M_{A5i}$ to $User_i$.

Step 5: After receiving M_{A6}, the user decrypts M_{A5i} with his/her private key K_i^{pr} and verifies the response V_4. After successful verification, SK is calculated as $SK = h(PDID_i \parallel N_i \parallel K_i) \oplus SK_i$. The user creates a new ID, $PDID_i^{new} = h(PDID_i \parallel K_i)$. The user stores the received nonce and the VC to use in the future. Thus, a session key SK is shared among all the participants. The authentication phase is illustrated in Table IV.

D. Key Recovery

Step 1: This step is required for key recovery if the private key is lost. The private key K_i^{pr} is divided into n shares with a (k, n) secret sharing scheme as explained in Section II. A polynomial of degree $k-1$ is selected as:

$$f(x) = K_i^{pr} + a_1(x) + \cdots + a_{k-1}(x)^{k-1}. \tag{5}$$

In (5), $a_1, a_2, \cdots, a_{k-1}$ are random polynomial coefficients. After that, the SP evaluates n values as $[K_i^{pr}]_1 = f(1)$, $[K_i^{pr}]_2 = f(2)$, $\cdots$, $[K_i^{pr}]_n = f(n)$. The shares of the private key calculated as mentioned above can be encrypted and stored in such a way that an adversary cannot correlate the shares.

Step 2: With any subset of k of these n values, the SP's private key can be calculated as

$$K_i^{pr} = \Sigma_{j=1}^{k} f(i_j) \Pi_{j \neq m} \frac{i_j}{i_j - i_m}. \tag{6}$$

With the knowledge of $k-1$ of these values, K_i^{pr} cannot be calculated. Note that during authentication, M_{A5i} is encrypted with K_i^{pu}. The user decrypts it with K_i^{pr}. If the private key is lost, it can be reconstituted from the backup using the (k, n) mechanism as mentioned in (6).

E. User Revocation

This phase can be initiated by the user or the SP.

Step 1: The user sends a request to the SP to get deregistered.

Step 2: Then, the SP removes the user from the accumulator by calling the function $Upd_{Acc}()$ as mentioned in Section II. After that, the SP removes the corresponding non-membership witness w_i of the user.

Note that the SP can also initiate the process of deregistering a user if required, e.g., if the user is dishonest.

V. Formal Security Proof

A. Definitions and assumptions

The proposed protocol's security is assessed using the Real-Or-Random (RoR) model [44]. Under the RoR model, we show that the proposed protocol can ensure session key security.

First, we will discuss the RoR model briefly. The protocol is secure if the established session key cannot be differentiated from a random string. In this model, security is defined through a series of games played between the participants and an adversary A. In our proof, we use imperative properties like a collision-resistant one-way hash function and ECDDHP. In the RoR model, we use the following queries to simulate the attacks.

Queries to model the attacks: During the authentication phase, the VCs of the user and the CS are verified by each other. The channel through which the authentication messages are exchanged is not secure. An adversary *A* can control the insecure channel between the user and the CS by eavesdropping and modifying the messages sent between them. Let us denote P_{EV}^t as the t^{th} instance of the EV and P_{CS}^t as the t^{th} instance of the CS. The following queries can be used to model these attack scenarios:

Execute(P_{EV}^{t1}, P_{CS}^{t2}): Models attacker *A*'s ability to eavesdrop and intercept the messages communicated between t_1^{th} instance of EV and t_2^{th} instance of CS in a session of the protocol.

Send(P^t, *m*): Models an active attack when the attacker impersonates a user or a CS and sends a message *m* to the oracle P^t, and then receives the response from P^t.

Reveal(P^t): Models *A*'s ability to obtain the session key SK established between P^t and its partner in a session of the protocol.

Test(P^t): The adversary executes this query to find the session key, and the output is either the session key SK or a random key based on an unbiased coin or hidden bit c. If $c = 1$, P^t returns SK. If $c = 0$, P^t returns a random number. Otherwise, P^t returns null.

A one-way hash function $h(\cdot)$ is also modelled as a random oracle. It is accessible to all the parties and A.

Theorem 1. Let A be an adversary trying to break the semantic security of the protocol.

A asks at most q_h hash queries. Let $|Hash|$ denote the hash output's length and let Adv_A^{ECDSA} represent A's advantage in breaking the ECDDHP problem. Then, the advantage of A in compromising the security of the session key in the proposed scheme is $Adv_A(t) \leq \frac{(q_h)^2}{|Hash|} + 2\, Adv_A^{ECDSA}$ which is negligible.

Proof: Let G_i denote a series of games where $i = 0, 1, \cdots, 2$. Let Adv_{A,G_i} denote A's advantage in the game G_i. Let $Success_A^{G_i}$ be the event when A correctly guesses the bit c in game G_i. Hence, $Adv_{A,G_i} = Pr[Success_A^{G_i}]$. We want the advantage of A to be negligible.

The games G_i where $i = 0, 1, 2$ are explained below:

Game G_0: This game corresponds to an actual attack by A against the proposed protocol. Since a bit c is selected randomly in G_0, we get:

$$Adv_A(t) = \mid 2Adv_{A,G_0} - 1 \mid . \quad (7)$$

Game G_1: In this game, A executes an eavesdropping attack in which A has the capability to intercept all the communicated messages. A calls the $Execute$ query to intercept the transmitted messages M_{A5} and M_{A6}. Then, A calls $Reveal$ and $Test$ queries to check if the captured session key is real or random. The session key SK is generated by the SP. SK_{CSj} and SK_i, which are transmitted through the messages M_{A5} and M_{A6}, respectively, are computed as $SK_i = h(PDID_i \parallel N_i \parallel K_i) \oplus SK$ and $SK_{CSj} = h(DID_{CSj} \parallel N_{CSj} \parallel K_{CSj}) \oplus SK$. A does not know the secret keys K_{CSj} and K_i to compute SK from SK_{CSj} and SK_i. To calculate the session key, A should also know the random nonce values N_{CSj} and N_i as well as the pseudo-identity of the user $PDID_i$ and the DID of the charging station DID_{CSj}. Hence, eavesdropping the messages M_{A5} and M_{A6} does not increase A's probability to win the game G_1. In other words, G_0 and G_1 are indistinguishable and

$$Adv_{A,G_1} = Adv_{A,G_0}. \quad (8)$$

Game G_2: In this game, A executes multiple $Send$ queries and tries to find a message digest collision. A one-way hash function is used for composing the messages M_{A3}, M_{A4}, M_{A5}, and M_{A6}. Further, in our protocol, the verifiable credentials change during each iteration of the protocol. They are signed by the SP with the private key using the ECDSA algorithm. It is a computationally infeasible problem to find the private key from the public key due to the intractability property of ECDDHP. Hence, knowing the public key of the SP also does not give any advantage to A. A should also know the other required secret parameters to compute the session key. Hence, from the birthday paradox of the hash function and intractability of ECDDHP, we get:

$$\mid Adv_{A,G_1} - Adv_{A,G_2} \mid \leq \frac{(q_h)^2}{2 \mid Hash \mid} + Adv_A^{ECDSA}(t). \quad (9)$$

After executing the above games, A guesses the bit c and calls the $Test$ query Then, we get the following:

$$Adv_{A,G_2} = \frac{1}{2}. \quad (10)$$

Combining (7) and (8), we get the following:

$$\begin{aligned} \frac{1}{2}Adv_A(t) &= \mid Adv_{A,G_0} - \frac{1}{2} \mid \\ &= \mid Adv_{A,G_1} - \frac{1}{2} \mid . \end{aligned} \quad (11)$$

From (9), (10), and (11), we have:

$$\begin{aligned} \frac{1}{2}Adv_A(t) &= \mid Adv_{A,G_1} - \frac{1}{2} \mid \\ &= \mid Adv_{A,G_1} - Adv_{A,G_2} \mid \\ &\leq \frac{(q_h)^2}{2 \mid Hash \mid} + Adv_A^{ECDSA}(t). \end{aligned} \quad (12)$$

By multiplying both sides of (12) by 2, we get:

$$Adv_A(t) \leq \frac{(q_h)^2}{\mid Hash \mid} + 2Adv_A^{ECDSA}(t). \quad (13)$$

Hence, this shows that the proposed protocol ensures session key security. ■

VI. Informal Security Analysis

In this section, we demonstrate how the proposed protocol achieves some of the important security properties for EV charging. We also discuss how the proposed protocol prevents the threats mentioned in the adversary model.

A. Security Properties

The key security features of the scheme are discussed below:

User-empowerment: The proposed scheme is based on DID. The users create and manage their IDs. Their IDs are not issued by any centralized issuing authority. The users are empowered by the fact that they have complete control over their IDs.

Effective User Revocation Policy: The proposed protocol incorporates an accumulator scheme that supports non-membership witnesses. The users must prove that they are not members of the revocation list by providing a non-membership witness issued by the SP. The user revocation policy enables the efficient removal of a user from the list of registered users when required. Since the revoked users are removed from the list of registered users, they cannot request EV charging.

Protection Against Unauthorized Use of Mobile Devices through Two-factor Security Verification: The user enters his/her biometrics (e.g., fingerprint) b_i and p_i to access MD_i. Then, MD_i compares $\delta_i^* = h(b_i \parallel p_i)$ with δ_i to verify $User_i$'s legitimacy. The remaining steps of the authentication phase will be carried out only if the biometric verification is successful. Even if an adversary gets the mobile device, he/she cannot use it for EV charging authentication as the biometric verification will fail. Thus, the proposed protocol provides protection against the unauthorized use of users' mobile devices.

Mutual Authentication: The charging station presents its VC to $User_i$ which $User_i$ verifies. Then, the user presents the ZKP of his/her VC to the CS and the CS verifies it. Only legitimate users and legitimate CSs hold valid VCs, signed with the private key of the SP K_{SP}^{pr}. Thus, both parties verify the VC of the other party, and the proposed protocol achieves mutual authentication.

Anonymity: Users reveal their real identities only to the SP. To charge the vehicle, the user uses a pseudo-ID $PDID_i$ and ZKP of a VC signed by the SP that the charging station verifies. The user's real identity is not revealed during the charging process. Thus, the proposed scheme ensures the anonymity of users.

Unlinkability: The user presents his/her pseudo-ID $PDID_i$ and ZKP of a verifiable credential to the CS during the authentication phase. For two consecutive sessions x and

$x+1$, $PDID_i^x \neq PDID_i^{x+1}$. Thus, the identities of the users are unlinkable.

Privacy of the User: The users use their pseudo-identities while charging their EVs. Additionally, the user provides a ZKP of the verified credential to the CS. The CS cannot learn anything about the user from the ZKP. Only the SP knows the user's real identity. The pseudo-identity and the verifiable credential of the user are changed during each session. For two consecutive sessions x and $x+1$, $PDID_i^x \neq PDID_i^{x+1}$. Hence, an adversary will be unable to link users' real identities with their pseudo-IDs or verifiable credentials. Thus, the proposed protocol ensures privacy.

Accountability: During registration, the SP verifies the VC of the digital identity of the user that is signed by a trusted organization, e.g., $[VC_{DigIDi}]_{K_{Gov}^{pr}}$. Thus, the SP confirms the legitimacy of the user to ensure accountability.

Session Key Agreement: During authentication, the SP generates a session key SK. The SP sends $SK_{CSj} = h(DID_{CSj} \parallel N_{CSj} \parallel K_{CSj}) \oplus SK$ and $V_3 = h(SK_{CSj} \parallel N_{CSj} \parallel K_{CSj})$ to the CS through M_{A5}. When the CS receives M_{A5}, SK is calculated as $SK = h(DID_{CSj} \parallel N_{CSj} \parallel K_{CSj}) \oplus SK_{CSj}$. Similarly, the SP sends $SK_i = h(PDID_i \parallel N_i \parallel K_i) \oplus SK$ and $V_4 = h(SK_i \parallel N_i \parallel K_i)$ to the user. When the user receives M_{A6}, SK is calculated as $SK = h(PDID_i \parallel N_i \parallel K_i) \oplus SK_i$. Thus, there is a session key agreement among the participants of the proposed protocol.

Non-Repudiation: After signing a statement with its private key, a party cannot deny having signed it (i.e., non-repudiation). The VC signing and verification are based on asymmetric cryptographic techniques. The SP signs the VC for the user and the CS with its private key (K_{SP}^{pr}). Only the SP knows the private key K_{SP}^{pr} to sign the VC. This guarantees non-repudiation.

Protection Against Impersonation Attacks: To send messages as a registered user $User_i$, the adversary must send $M_{A3} : \{hV, \pi, N_i, EL_i, V_1, w_i\}$ to the CS. EL_i and V_1 have to be computed to compose M_{A3}. Only $User_i$ knows the parameter K_i to calculate $EL_i = L_i \oplus h(K_i \parallel N_i)$ and $V_1 = h(PDID_i \parallel N_i \parallel K_i \parallel EL_i)$. Hence, an adversary's attempt to send messages as a registered user will not be successful. Similarly, to send messages as a registered CS CS_j, the adversary must generate $M_{A4} : \{M_{A3}, DID_{CSj}, N_{CSj}, L_{CSj}, V_2\}$. Only CS_j has the knowledge of the parameter K_{CSj} to compute $V_2 = h(DID_{CSj} \parallel N_{CSj} \parallel K_{CSj} \parallel L_{CSj})$. To send messages as the SP, the attacker needs to have the knowledge of K_{CSj} and K_i to compute $V_3 = h(SK_{CSj} \parallel N_{CSj} \parallel K_{CSj})$ and V_4 as $V_4 = h(SK_i \parallel N_i \parallel K_i)$. The adversary does not have the knowledge of K_{CSj} and K_i to generate valid responses. Thus, the proposed protocol is robust against impersonation attacks.

Protection Against Location Forgery Attacks: The EV charging price depends on the charging station's location. A dishonest user or a dishonest charging station may present a false location identifier to the SP. When the SP receives message M_{A4}, it decodes L_i from EL_i and compares L_i with L_{CSj}. If the comparison of location identities fails, the authentication process will be terminated. Thus, the proposed authentication protocol prevents location forgery attacks.

Protection Against Replay Attacks: An adversary may capture the messages transmitted between different entities and replay them later to get authenticated. To prevent it, N_i in $M_{A3} : \{hV, \pi, N_i, EL_i, V_1, w_i\}$ is not repeated in two sessions of the proposed protocol. Similarly, N_{CSj} in $M_{A4} : \{M_{A3}, DID_{CSj}, N_{CSj}, L_{CSj}, V_2\}$ is not repeated. Hence, an adversary's attempt to replay M_{A3} and M_{A4} will not be successful. Similarly, the responses V_3 and V_4 from the SP are computed using the values N_{CSj} and N_i that are not repeated. This ensures that the adversary's attempt to replay the captured messages M_{A5} and M_{A6} from the SP will not be successful since V_3 and V_4 are used to compose M_{A5} and M_{A6}, respectively. Thus, the proposed scheme prevents replay attacks.

VII. Performance Analysis and Comparison

This section compares the proposed scheme's security features with other EV charging schemes in the literature. Then, we evaluate the proposed protocol's computation cost. After that, we compare the proposed protocol with other existing protocols based on the computation cost.

V2G communications must be secure and efficient. We have shown that the proposed protocol is secure through formal and informal security analyses. To make the proposed protocol efficient, we have tried to minimize the required computation resources wherever possible. As an example, we have used the Zero-Knowledge proof (ZKP) of the VCs only for users as our focus is on the privacy of users. The charging stations present their VCs directly without generating a ZKP. By not using the ZKP of the VCs of the charging stations, we have reduced the ZKP generation step at the charging station and the ZKP verification step at the user. To improve the performance, another aspect we have considered is using an accumulator scheme with non-membership witnesses. Since the number of revoked users is usually less than the number of users who register, the accumulator needs to be updated less often, and hence it is more efficient to use an accumulator scheme with non-membership witnesses than it is to use other accumulators. Also, the registration phase is executed only once which further reduces the computation cost.

A. Comparison of Security Features

A comparison of the security features is given in Table V. The main feature that sets our protocol apart from other protocols is user-empowerment. We leveraged DID to achieve user empowerment. Also, while most similar works ensure privacy and anonymity for the user, they do not provide non-repudiation. The proposed scheme ensures non-repudiation. The proposed scheme also prevents location forgery attacks. Most of the other schemes do not protect against location forgery attacks. A user-empowered authentication protocol built on DID and VC was proposed for EV charging in [30]. A blockchain-based energy trading scheme for vehicle-to-vehicle communications based on DIDs was proposed in [31]. However, the protocols in [30] and [31] are vulnerable to

TABLE V
COMPARISON BASED ON SECURITY FEATURES

Scheme	SF1	SF2	SF3	SF4	SF5	SF6	SF7	SF8	SF9	SF10	SF11
Roman et al. [1]	Yes	Yes	Yes	Yes	Yes	Yes	No	No	-	-	No
Zhang et al. [10]	Yes	Yes	Yes	Yes	Yes	No	No	No	Yes	-	No
Bansal et al. [16]	Yes	Yes	Yes	Yes	Yes	No	No	No	-	-	No
Gope et al. [19]	Yes	Yes	Yes	Yes	Yes	Yes	No	No	Yes	-	No
Saxena et al. [24]	Yes	Yes	Yes	Yes	Yes	No	No	No	-	-	Yes
Parameswarath et al. [30]	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	No	No	No
Kim et al. [31]	Yes	Yes	Yes	Yes	Yes	No	Yes	Yes	-	No	No
$Proposed\ Scheme$	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
SF1: Privacy of the user; **SF2:** Anonymity of the user; **SF3:** Session key;											
SF4: Protection against replay attacks; **SF5:** Mutual authentication;											
SF6: Protection against location forgery attacks; **SF7:** Non-repudiation; **SF8:** User-empowerment;											
SF9: Two-factor security verification ; **SF10:** Usability/Key recovery; **SF11:** User revocation											

TABLE VI
COMPUTATION COSTS OF VARIOUS SCHEMES

Scheme	User's Device	SP/CS
Roman et al. [1]	$3M + B$ = 26.13 ms	$16M + 3B$ = 86.84 ms
Zhang et al. [10]	$3M + EXP + 5H$ = 24.58 ms	$3M + B + 4H$ = 30.69 ms
Bansal et al. [16]	$1P + 2MAC + 4EXP + 2S + 2M$ = 31.874 ms	$1P + 6MAC + 4S + 6EXP + 4M$ = 65.814 ms
Gope et al. [19]	$6H$ = 6.84 ms	$8H$ = 9.12 ms
Saxena et al. [24]	$1M + 3EXP + 1B$ = 25.13 ms	$5M + 8EXP + 1B$ = 61.89 ms
Parameswarath et al. [30]	$8H + Enc_{RSA} + Dec_{RSA} + Verify_{RSA}$ = 41.42 ms	$10H + Enc_{RSA} + Dec_{RSA} + Verify_{RSA} + Sign_{RSA}$ = 66.4 ms
Kim et al. [31]	$5H + 5MP + 2AP + S$ = 23.94 ms	$5H + 5MP + 2AP + S$ = 23.94 ms
$Proposed\ Scheme$	$6H + Verify_{ECDSA} + Prove_{ZKP}$ = 28.06 ms	$8H + Sign_{ECDSA} + Verify_{ZKP}$ = 45.95 ms
M**:** Multiplication Operation = 5.24 ms; MAC**:** MAC Operation = 1.3 ms; AP**:** Elliptic Curve Point Addition Operation = 1.01 ms;		
P**:** PUF Operation = 0.014 ms;H**:** Hash Operation = 1.14 ms ; MP**:** Multiplication Point Operation = 3.21 ms;		
EXP**:** Modular Exponential Operation = 3.16 ms; S**:** Symmetric Encryption/Decryption Operation = 0.17 ms;		
B**:** Bilinear Pairing Operation = 10.41 ms; $Sign_{ECDSA}$**:** ECDSA Signing = 14.31 ms; $Verify_{ECDSA}$**:** ECDSA Signature Verification = 16.7 ms		
Enc_{RSA}**:** RSA Encryption = 3.6 ms; Dec_{RSA}**:** RSA Decryption = 4.2 ms; $Sign_{RSA}$**:** RSA Signing = 22.7 ms		
$Verify_{RSA}$**:** RSA Signature Verification = 28.6 ms ; $Prove_{ZKP}$**:** ZKP Generation = 4.52 ms; $Verify_{ZKP}$**:** ZKP Verification = 1.3 ms		

the usability problem where an adversary deletes the user's private key before being noticed. The proposed protocol provides an option for key recovery in such a scenario which is not provided by [30] and [31]. The protocol in [30] is also vulnerable to the unauthorized use of the mobile device of a legitimate user for EV authentication by an adversary. The proposed protocol has a biometric verification step where users must input their biometric details to access the mobile device. This step prevents the unauthorized use of the mobile device. This security protection is not offered by [30]. The proposed protocol also provides an effective user revocation policy to remove users if required. User revocation is not addressed in [30] and [31]. Hence, the proposed scheme's security features are better than those of other schemes.

B. Analysis of Computation Cost

Next, we evaluate the computation cost of the proposed scheme and compare it with the cost of other similar schemes. We used a personal computer with an Intel Core i5 CPU, 2.90 GHz clock, and 8 GB of RAM to execute the cryptographic operations. For the ZKP system, we used the API implementation given in [45]. The time taken by various cryptographic operations is given in Table VI.

The registration process is executed only once. However, the authentication process is executed each time the user wants to charge the vehicle. Hence, we discuss the computation cost during the authentication phase of the protocol in this section while omitting the computation cost during the initial registration. Since the time taken by the XOR operation and concatenation operation is negligible, the time taken by these operations is not considered when evaluating the computation cost of the protocols. To calculate the computation cost of the proposed scheme, we counted the number of cryptographic operations executed during the authentication phase and used their execution time. During the authentication phase, the user's device executes 6 hash, one ECDSA verification, and one ZKP prove operations. The service provider and charging station together execute 8 hash, one ECDSA signing, and one ZKP verification operation. Hence, the total computation costs at the EV and SP/CS are $6H + Verify_{ECDSA} + Prove_{ZKP} = 28.0$ ms and $8H + Sign_{ECDSA} + Verify_{ZKP} = 45.95$ ms, respectively. Similarly, we counted the number of cryptographic operations in other schemes and calculated the computation costs. The computation costs of these schemes are presented in Table VI.

Next, we compare the computation cost of the proposed scheme with other schemes. Figure 5 shows a comparison of the computation time taken by the user's device during authentication in different schemes. Figure 6 and Figure 7 show the computation time taken by the CS/SP and the total time taken during authentication in different schemes. We

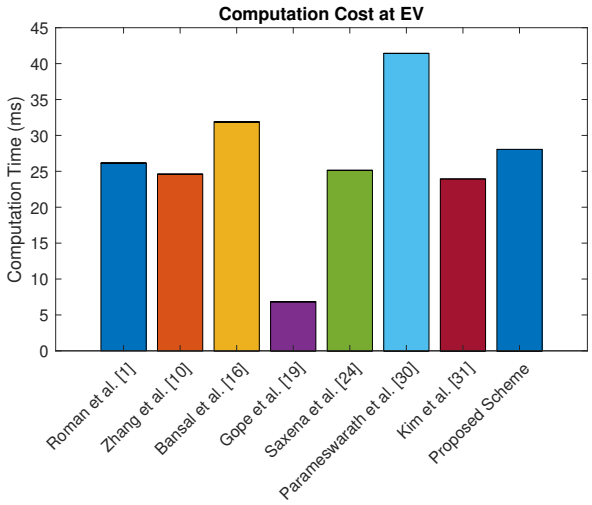

Fig. 5. Comparison of computation cost at EV.

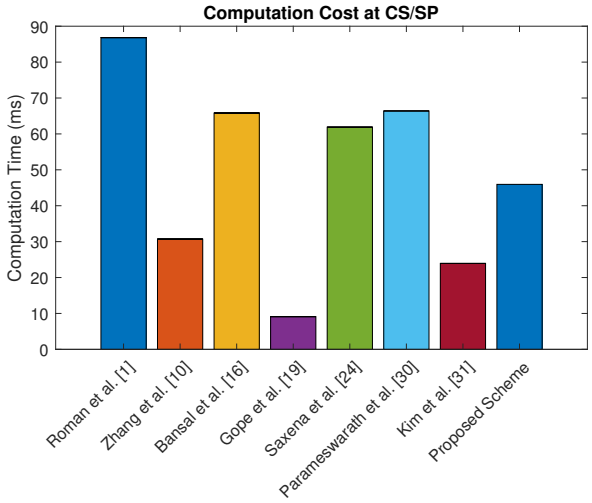

Fig. 6. Comparison of computation cost at CS/SP.

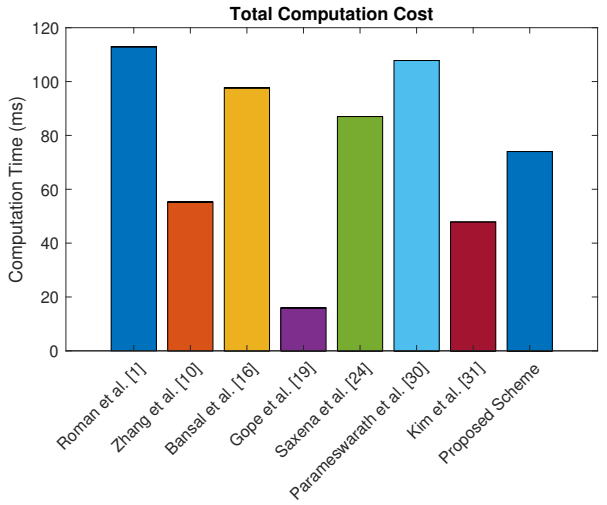

Fig. 7. Comparison of total computation cost.

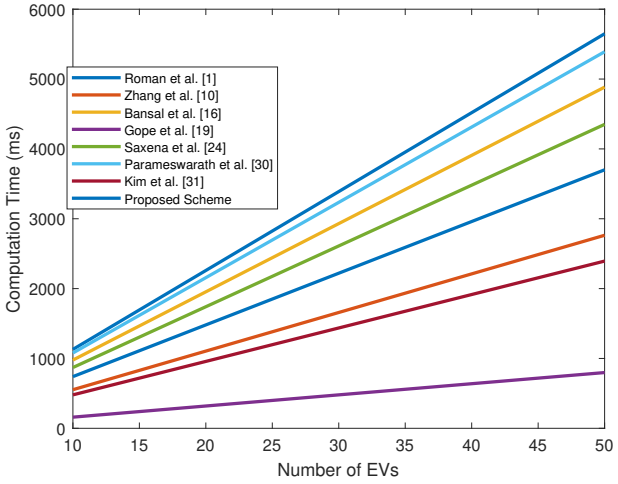

Fig. 8. Computation cost as a function of the number of EVs.

now examine the effect of the number of EVs on computation time. The computation cost as a function of the number of EVs is plotted in Figure 8. When a user's VC is verified, the SP sends the user a new VC for the next round. This step is required to provide unlinkability between the VCs in consecutive sessions. The computation cost at the SP in the proposed scheme is slightly higher than that in some other schemes since generating a new VC adds to the computation cost at the SP in the proposed scheme. Considering the fact that the SP has sufficient resources to do the computation and the security properties the proposed scheme offers, the computation cost at the SP is reasonable. As a result, we can conclude that the proposed scheme has a reasonable computation time.

VIII. Conclusion

In this paper, we presented a DID, VC, and ZKP-based authenticated key exchange protocol for V2G communication that allows users to have complete ownership of their IDs. They can charge their electric vehicles without revealing their identities. The user presents the ZKP of the VC to prove his/her legitimacy before charging the vehicle. By making use of ZPK, the user does not have to reveal anything other than the fact that he/she is a legitimate user. The proposed protocol also provides an option to recover the private key of the user in the event it is lost. The key recovery mechanism makes the proposed protocol resilient to the accidental loss of a private key. The proposed protocol also provides an effective method for user revocation that can be initiated by the user or by the SP. We provided informal and formal security analyses to show that the proposed mechanism is robust. The proposed mechanism achieves security properties such as session key security, mutual authentication, privacy, anonymity, unlinkability, and protection against several attacks. We compared the proposed protocol with other similar works regarding performance and security properties. Our analysis shows that the proposed protocol offers all major security features at a reasonable computational cost.

IX. Acknowledgment

This work was supported in part by the Ministry of Education, Singapore under grants A-0009040-00-00 and A-0009040-01-00.

References

[1] L. F. Roman, P. R. Gondim, and J. Lloret, "Pairing-based authentication protocol for v2g networks in smart grid," *Ad Hoc Networks*, vol. 90, p. 101745, 2019.

[2] Iea (2021), global ev outlook 2021, iea, paris. [Online]. Available: https://www.iea.org/reports/global-ev-outlook-2021

[3] H. Li, D. Han, and M. Tang, "A privacy-preserving charging scheme for electric vehicles using blockchain and fog computing," *IEEE Systems Journal*, 2020.

[4] J. Antoun, M. E. Kabir, B. Moussa, R. Atallah, and C. Assi, "A detailed security assessment of the ev charging ecosystem," *IEEE Network*, vol. 34, no. 3, pp. 200–207, 2020.

[5] J. Krumm, "Inference attacks on location tracks," in *International Conference on Pervasive Computing*, vol. 4480 LNCS. Microsoft Research, 2007, pp. 127–143.

[6] C. Allen, "The Path to Self-Sovereign Identity ," Online,http://www.lifewithalacrity.com/2016/04/the-path-to-self-soverereign-identity.html, [Accessed: Nov 2022].

[7] "Decentralized Identifiers (DIDs)," Online,https://www.w3.org/TR/did-core/, [Accessed: Aug 2022].

[8] A. Unterweger, F. Knirsch, D. Engel, D. Musikhina, A. Alyousef, and H. de Meer, "An analysis of privacy preservation in electric vehicle charging," *Energy Informatics*, vol. 5, no. 1, pp. 1–27, 2022.

[9] 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 Transactions on Information Forensics and Security*, vol. 9, no. 2, pp. 208–220, 2014.

[10] Y. Zhang, J. Zou, and R. Guo, "Efficient privacy-preserving authentication for v2g networks," *Peer-to-Peer Networking and Applications*, vol. 14, no. 3, pp. 1366–1378, 2021.

[11] L. Chen, J. Zhou, Y. Chen, Z. Cao, X. Dong, and K.-K. R. Choo, "Padp: Efficient privacy-preserving data aggregation and dynamic pricing for vehicle-to-grid networks," *IEEE Internet of Things Journal*, vol. 8, no. 10, pp. 7863–7873, 2021.

[12] S. Gunukula, A. B. T. Sherif, M. Pazos-Revilla, B. Ausby, M. Mahmoud, and X. S. Shen, "Efficient scheme for secure and privacy-preserving electric vehicle dynamic charging system," in *2017 IEEE International Conference on Communications (ICC)*, 2017, pp. 1–6.

[13] N. Saxena and B. J. Choi, "Authentication scheme for flexible charging and discharging of mobile vehicles in the v2g networks," *IEEE Transactions on Information Forensics and Security*, vol. 11, no. 7, pp. 1438–1452, 2016.

[14] A. Abdallah and X. S. Shen, "Lightweight authentication and privacy-preserving scheme for v2g connections," *IEEE Transactions on Vehicular Technology*, vol. 66, no. 3, pp. 2615–2629, 2017.

[15] M. Kaveh, D. Martín, and M. R. Mosavi, "A lightweight authentication scheme for v2g communications: A puf-based approach ensuring cyber/physical security and identity/location privacy," *Electronics*, vol. 9, no. 9, p. 1479, 2020.

[16] G. Bansal, N. Naren, V. Chamola, B. Sikdar, N. Kumar, and M. Guizani, "Lightweight mutual authentication protocol for v2g using physical unclonable function," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 7, pp. 7234–7246, 2020.

[17] W. Hou, Y. Sun, D. Li, Z. Guan, and J. Liu, "Lightweight and privacy-preserving charging reservation authentication protocol for 5g-v2g," *IEEE Transactions on Vehicular Technology*, 2023.

[18] V. T. Kilari, S. Misra, and G. Xue, "Revocable anonymity based authentication for vehicle to grid (v2g) communications," in *2016 IEEE International Conference on Smart Grid Communications (SmartGridComm)*. IEEE, 2016, pp. 351–356.

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

[20] A. Irshad, M. Usman, S. A. Chaudhry, H. Naqvi, and M. Shafiq, "A provably secure and efficient authenticated key agreement scheme for energy internet-based vehicle-to-grid technology framework," *IEEE Transactions on Industry Applications*, vol. 56, no. 4, pp. 4425–4435, 2020.

[21] S. Aggarwal, N. Kumar, and P. Gope, "An efficient blockchain-based authentication scheme for energy-trading in v2g networks," *IEEE Transactions on Industrial Informatics*, vol. 17, no. 10, pp. 6971–6980, 2020.

[22] Y. Su, G. Shen, and M. Zhang, "A novel privacy-preserving authentication scheme for v2g networks," *IEEE Systems Journal*, vol. 14, no. 2, pp. 1963–1971, 2020.

[23] K. Kaur, S. Garg, G. Kaddoum, F. Gagnon, S. H. Ahmed, and M. Guizani, "A secure, lightweight, and privacy-preserving authentication scheme for v2g connections in smart grid," in *IEEE INFOCOM 2019-IEEE conference on computer communications workshops (INFOCOM WKSHPS)*. IEEE, 2019, pp. 541–546.

[24] N. Saxena, B. J. Choi, and S. Cho, "Lightweight privacy-preserving authentication scheme for v2g networks in the smart grid," in *2015 IEEE Trustcom/BigDataSE/ISPA*, vol. 1. IEEE, 2015, pp. 604–611.

[25] Y. Kortesniemi, D. Lagutin, T. Elo, and N. Fotiou, "Improving the privacy of iot with decentralised identifiers (dids)," *Journal of Computer Networks and Communications*, vol. 2019, 2019.

[26] M. Eisenstadt, M. Ramachandran, N. Chowdhury, A. Third, and J. Domingue, "Covid-19 antibody test/vaccination certification: there's an app for that," *IEEE Open Journal of Engineering in Medicine and Biology*, vol. 1, pp. 148–155, 2020.

[27] H. ElHusseini, C. Assi, B. Moussa, R. Attallah, and A. Ghrayeb, "Blockchain, ai and smart grids: The three musketeers to a decentralized ev charging infrastructure," *IEEE Internet of Things Magazine*, vol. 3, no. 2, pp. 24–29, 2020.

[28] Z. Su, Y. Wang, Q. Xu, M. Fei, Y.-C. Tian, and N. Zhang, "A secure charging scheme for electric vehicles with smart communities in energy blockchain," *IEEE Internet of Things Journal*, vol. 6, no. 3, pp. 4601–4613, 2018.

[29] X. Li, T. Jing, R. Li, H. Li, X. Wang, and D. Shen, "Bdra: Blockchain and decentralized identifiers assisted secure registration and authentication for vanets," *IEEE Internet of Things Journal*, 2022.

[30] R. P. Parameswarath, P. Gope, and B. Sikdar, "User-empowered privacy-preserving authentication protocol for electric vehicle charging based on decentralized identity and verifiable credential," *ACM Transactions on Management Information Systems (TMIS)*, 2022. [Online]. Available: https://doi.org/10.1145/3532869

[31] M. Kim, J. Lee, J. Oh, K. Park, Y. Park, and K. Park, "Blockchain based energy trading scheme for vehicle-to-vehicle using decentralized identifiers," *Applied Energy*, vol. 322, p. 119445, 2022.

[32] A. Shamir, "How to share a secret," *Communications of the ACM*, vol. 22, no. 11, pp. 612–613, 1979.

[33] Own your digital identity. [Online]. Available: https://www.microsoft.com/en-sg/security/business/identity-access-management/decentralized-identity-blockchain

[34] J. Groth, "On the size of pairing-based non-interactive arguments," in *Annual international conference on the theory and applications of cryptographic techniques*. Springer, 2016, pp. 305–326.

[35] M. de Vasconcelos Barros, F. Schardong, and R. Felipe Custódio, "Leveraging self-sovereign identity, blockchain, and zero-knowledge proof to build a privacy-preserving vaccination pass," *Blockchain, and Zero-Knowledge Proof to Build a Privacy-Preserving Vaccination Pass*, 2022.

[36] X. Salleras and V. Daza, "Zpie: Zero-knowledge proofs in embedded systems," *Mathematics*, vol. 9, no. 20, p. 2569, 2021.

[37] N. Koblitz, "Elliptic curve cryptosystems," *Mathematics of computation*, vol. 48, no. 177, pp. 203–209, 1987.

[38] Y. Choi, D. Lee, J. Kim, J. Jung, J. Nam, and D. Won, "Security enhanced user authentication protocol for wireless sensor networks using elliptic curves cryptography," *Sensors (Switzerland)*, vol. 14, no. 6, pp. 10 081 – 10 106, 2014. [Online]. Available: http://dx.doi.org/10.3390/s140610081

[39] S. D. Galbraith and P. Gaudry, "Recent progress on the elliptic curve discrete logarithm problem," *Des. Codes Cryptography*, vol. 78, no. 1, p. 51–72, jan 2016. [Online]. Available: https://doi.org/10.1007/s10623-015-0146-7

[40] Ecdsa: Elliptic curve signatures. [Online]. Available: https://cryptobook.nakov.com/digital-signatures/ecdsa-sign-verify-messages

[41] S. M. Smith, "Key event receipt infrastructure (keri)," 2021. [Online]. Available: https://arxiv.org/abs/1907.02143

[42] J. Camenisch and A. Lysyanskaya, "Dynamic accumulators and application to efficient revocation of anonymous credentials," in *Annual international cryptology conference*. Springer, 2002, pp. 61–76.

[43] J. Li, N. Li, and R. Xue, "Universal accumulators with efficient nonmembership proofs," in *International Conference on Applied Cryptography and Network Security*. Springer, 2007, pp. 253–269.

[44] M. Abdalla, P.-A. Fouque, and D. Pointcheval, "Password-based authenticated key exchange in the three-party setting," in *International workshop on public key cryptography*. Springer, 2005, pp. 65–84. [Online]. Available: https://doi.org/10.1007/978-3-540-30580-4_6

[45] X. Salleras and V. Daza, "Zpie: Zero-knowledge proofs in embedded systems," *Mathematics*, vol. 9, no. 20, p. 2569, 2021.

Rohini Poolat Parameswarath received the Master of Technology degree in Software Engineering from the National University of Singapore, Singapore in 2009. She is a cyber security researcher at the Department of Electrical and Computer Engineering, National University of Singapore, Singapore. Currently, she is pursuing a PhD with research focusing on protocols for security and privacy in vehicular environments. Prior to joining the National University of Singapore, she was part of the cyber security research team at the Singapore University of Technology and Design, Singapore. Before embarking on her career in cyber security research, she worked as a software engineer in multinational companies. She is passionate about finding solutions to the current challenges in the cybersecurity landscape. Her research interests include cyberattack detection, ways to prevent attacks, data privacy, and cryptographic protocols in domains such as the Internet of Things (IoT), cyber-physical systems, and vehicular networks. Her papers have been published in top-tier conferences and journals such as IEEE GLOBECOM, IEEE VTC, IEEE Internet of Things Magazine, ACM Transactions on Management Information Systems, and more.

Prosanta Gope (Senior Member, IEEE) was a Research Fellow with the Department of Computer Science, National University of Singapore. He is currently an Assistant Professor at the Department of Computer Science (Cyber Security), The University of Sheffield, U.K. He has authored more than 100 peer-reviewed papers in several reputable international journals and conferences and has four filed patents. His papers have been published in high-impact journals, such as IEEE Transactions on Information Forensics and Security, IEEE Transactions on Dependable and Secure Computing, IEEE Transactions on Industrial Electronics, and IEEE Transactions on Smart Grid. Primarily driven by challenging real-world security problems, he has expertise in lightweight anonymous authentication, authenticated encryption, access control, mobile communications security, healthcare, the Internet of Things, Cloud, RFIDs, WSNs, smart grids, and hardware security of the IoT devices. He has served as the TPC Member/Chair for several reputable international conferences, such as ESORICS, IEEE TrustCom, and ARES. He is an Associate Editor of the IEEE Internet of Things Journal, IEEE Systems Journal, IEEE Sensors Journal, and the Journal of Information Security and Applications (Elsevier).

Biplab Sikdar (Senior Member, IEEE) is a Professor in the Department of Electrical and Computer Engineering at the National University of Singapore, where he also serves as the Interim Head of the Department of Electrical and Computer Engineering. He received the B. Tech. degree in electronics and communication engineering from North Eastern Hill University, Shillong, India, in 1996, the M.Tech. degree in electrical engineering from the Indian Institute of Technology, Kanpur, India, in 1998, and the Ph.D. degree in electrical engineering from the Rensselaer Polytechnic Institute, Troy, NY, USA, in 2001. He was an Assistant Professor from 2001-2007 and Associate Professor from 2007-2013 in the Department of Electrical, Computer, and Systems Engineering at Rensselaer Polytechnic Institute from 2001 to 2013. He is a recipient of the NSF CAREER award, the Tan Chin Tuan fellowship from NTU Singapore, the Japan Society for Promotion of Science fellowship, and the Leiv Eiriksson fellowship from the Research Council of Norway. His research interests include IoT and cyber-physical system security, network security, and network performance evaluation. Dr. Sikdar is a member of Eta Kappa Nu and Tau Beta Pi. He has served as an Associate Editor for the IEEE Transactions on Communications, IEEE Transactions on Mobile Computing, and IEEE Internet of Things Journal and is an IEEE COMSOC and VTS Distinguished Lecturer and ACM Distinguished Speaker.