Optimizing V2G Dynamics: An AI-Enhanced Secure Protocol for Energy Management in Industrial Cyber-Physical Systems

Shafiq Ahmed, Mohammad Hossein Anisi, *Senior Member, IEEE*

***Abstract*—The rapid advancement of intelligent transportation systems and the growing demand for sustainable energy solutions have elevated the Vehicle-to-Grid (V2G) paradigm in Industrial Cyber-Physical Systems (ICPS). This paper presents an AI-Enhanced Secure Protocol for V2G Energy Management, integrating Artificial Intelligence (AI) through Long Short-Term Memory (LSTM) networks with advanced cryptographic techniques for optimizing energy distribution between smart grids and electric vehicles. This protocol enhances system security and device integrity, effectively countering cyber threats and physical tampering. Emphasizing practical applicability, it demonstrates scalability and versatility across various smart grid environments, marking a significant step in AI-integrated cybersecurity for sustainable energy management. Comparative analysis reveals reductions in computation and communication costs by 49.79% and 23.24%, respectively, highlighting the efficiency of the protocol and its potential to enhance smart grid security frameworks.**

***Index Terms*—Security, Electric Vehicles, Vehicle to Grid, ICPS, Smart Grid**

I. Introduction

ELECTRIC vehicles (EVs) are pivotal in Vehicle-to-Grid (V2G) networks, part of Industrial Cyber-Physical Systems, for energy storage without extra hardware investment. Their batteries enable parked or idle EVs to function as flexible energy sources, allowing energy trading with the grid. This bidirectional energy flow enhances electricity generation and grid quality, reducing peak demand and emissions [1], [2]. However, the integration of telecommunication technologies in V2G systems introduces vulnerabilities such as replay and denial of service attacks [3], necessitating enhanced System Security and Embedded Device Security.

Existing V2G authentication protocols using cryptographic methods face challenges in resource-limited devices or lack certain security features [4]–[11]. We propose an AI-Enhanced Secure Protocol for V2G Energy Management in Industrial Cyber-Physical Systems, balancing robust security with efficient resource use.

The integration of vehicles into smart grids, initiated by Kempton and Tomić in 2004 [12], has driven significant research into secure and efficient smart grid protocols [13], [14].

Mohammadali et al. [15] introduced an identity-based protocol minimizing computational demands and resisting "replay" and "desynchronization" attacks but failed against "impersonation," "false data injection," and "man-in-the-middle attacks." Nicanfar et al.'s protocol [16] also struggles with high computational load and "false data injection attacks." Wu et al. proposed a more robust alternative combining symmetric and asymmetric cryptographic methods [16], [17].

Tsai and Lo [4] introduced a key distribution method using bilinear cryptography, emphasizing mutual authentication and conditional privacy. Odelu et al. [5] proposed an enhanced key agreement protocol addressing its shortcomings in session key protection and computational demands.

Gope and Sikdar [6] identified vulnerabilities in Odelu et al.'s protocol, offering a physically secure alternative using $PUFs$. Irshad et al. [7] proposed a more secure key agreement protocol for smart grids.

Zhan and Yu [18] introduced a lightweight protocol using one-way non-collision hash functions, focusing on physical security. Badar et al. [9] developed a protocol for power line surveillance, enhancing data transfer security but facing challenges with secret parameter protection and desynchronization attacks.

Gope et al. [10] introduced a reconfigurable authentication and key agreement system using $PUFs$ to combat electricity theft, lacking user access revocation and dynamic addition functionalities.

Their later work [11] presented an anonymous, lightweight IoT authentication protocol using physical unclonable capabilities to enhance security and reduce computational load, addressing wireless data transmission challenges.

Modarres et al. [19] identified vulnerabilities in Gope et al.'s IoT protocol, including susceptibility to DOS and replay attacks, and a lack of forward secrecy. These findings highlight areas for improvement in IoT security protocols.

Considering the limitations of existing authentication protocols, which often compromise on security features or incur high computational costs, this paper proposes an AI-Enhanced Secure Protocol for V2G interactions in smart grids. Designed to address these challenges, the protocol combines robust security with sustainable computational and communication efficiency, aiming to fill the existing security gaps in smart grid environments.

A. Motivation and Contribution

The increasing demand for sustainable energy management and intelligent transportation systems has highlighted

S. Ahmad and M. H. Anisi are with the School of Computer Science and Electronic Engineering, University of Essex, Colchester CO4 3SQ, UK. E-mail: sa23281@essex.ac.uk; m.anisi@essex.ac.uk

TABLE I
SUMMARY OF EXISTING RELATED WORK

Authors	Year	Advantage(s)	Limitation(s)
[4]	2015	Provides mutual authentication & conditional privacy	Not providing confidentiality
[5]	2016	Demands less computational power	Susceptible to man in the middle and denial of service attack.
[7]	2020	Provides fairly secure key agreement protocol for smart grid environment	Does not offer anonymity and untraceabilty
[8]	2020	Providing protection for smart meters	Vulnerable to physical attacks
[9]	2020	Provides smart meter surveillance	Prone to desynchronization attack
[10]	2021	Introduced re-configurable authentication and key agreement using PUF	Lack of user access revocation & Fuzzy extrator used in PUF has computational limitations
[11]	2021	Introduced anonymous and lightweight authentication system for IoT using PUF	Susceptible to DOS and replay attacks.
[20]	2022	Proposed system provides security and trustworthiness	Higher computation and communication cost limits the application in resource constrained environment
[21]	2021	Unique Edge computing based architecture used	Failed to provide privacy protection.
[22]	2021	Signcription based AKA protocol proposed which was novel at that time	Susceptible to physical attacks.
[23]	2023	Lightweight & comprehensively designed protocol	Vulnerable to physical attacks as well as impersonation attacks

TABLE II
NOTATIONS AND THEIR MEANINGS

Notation	Meaning
V_{id}, V_{pwd}	ID and password of EV
CS_{id}	ID of Charging Station
K_{cs}, K_v	CS and EV Secret Keys
r_v, r_{cs}, r_e	Randomly Generated Numbers
PID_v	Pseudo ID of Vehicle
LS_v, LS_{cs}	Location Service of EV and CS
α_a, α_b	Challenge Messages
β_a, β_b	Response against challenge Messages
W_{PUF}, R_{PUF}	Weak PUF and Reconfigurable PUF
$h(.)$	Collision Resistant One Way Hash Function
τ	Threshold Value
$\oplus, \|\|$	Bitwise Xor and Concatenation Operation
Adv	Adversary
SK	Session Key

the Vehicle-to-Grid (V2G) concept within Industrial Cyber-Physical Systems (ICPS). Addressing the limitations of current V2G protocols, particularly high computation costs and inadequate physical security, we propose an AI-Enhanced Secure Protocol. This protocol leverages AI techniques and cryptographic methods, emphasizing the often-overlooked role of Physical Unclonable Functions (PUF) in enhancing V2G security.

We establish a tailored system and attack model for V2G communication to ensure secure and efficient authentication. Our PUF-enabled, identity-based authentication protocol is lightweight and effectively mitigates known physical security threats. By utilizing basic cryptographic operations with PUF technology, we create a low-cost, highly secure protocol. Rigorous security analyses, both formal and informal, demonstrate the protocol's effectiveness against potential threats. Performance evaluations confirm our protocol's superior computational and communication efficiency over existing models, significantly advancing secure, sustainable energy management for ICPS.

II. PRELIMINARIES

This section outlines the key system models and cryptographic primitives for understanding the proposed protocol. Table II lists the notations and their full forms.

A. Physical Unclonable Function

A Physical Unclonable Function (PUF) is a one-way function using a device's physical characteristics to map challenges to responses [24]. PUFs have several key attributes [25]: The output is influenced by the device's physical design. PUFs are easy to analyze and implement. Outputs are unpredictable and behave like random functions. PUFs generate unique outputs despite identical configurations. Additionally, PUFs are unclonable and maintain unique identities.

In our AI-Enhanced Secure Protocol for V2G Energy Management, PUFs are essential for security. They uniquely identify devices, resisting cloning and emulation attacks, thus ensuring network integrity and authenticity. By integrating PUFs, we add robust protection against tampering and cyber threats, enhancing the V2G ecosystem's security and efficiency. Our protocol uses two PUF variants: strong and weak PUFs.

B. Modeling Attacks

The application of machine learning techniques has exposed realistic $PUFs$ to increased susceptibility to modeling attacks. To execute such an attack, an adversary is required to amass a substantial dataset of Challenge-Response Pairs (CRPs) denoted as $(\alpha_a, \beta_a), (\alpha_b, \beta_b), \ldots, (\alpha_n, \beta_n)$. Utilizing this dataset, the adversary endeavours to develop a mathematical model, represented as S, which encapsulates the behavioural patterns of the PUF. This model is then employed to predict the response β_{n+1} of the PUF to a novel challenge α_{n+1}. The extensive volume of CRPs required to adequately train this model renders Strong PUFs particularly vulnerable to such modeling attacks [26].

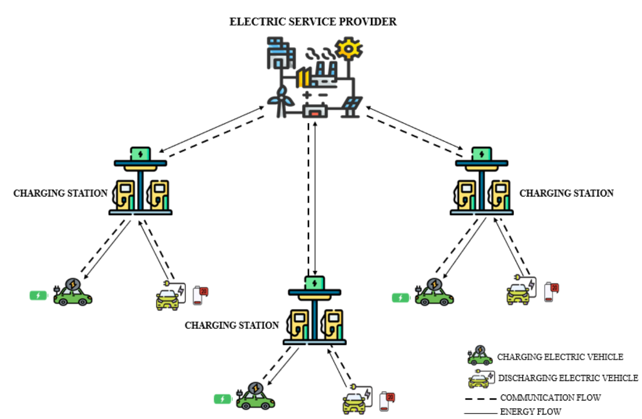

Fig. 1. Proposed Network Model

C. Network Model

As shown in Figure 1, the smart grid network includes the ESP (Energy Service Provider), CS (Charging Station), and

EV (Electric Vehicle). The ESP manages power generation, distribution, and the database of registered EVs and CSs. CSs provide charging services to EVs, powered by the ESP. EVs, with secure On-Board Units (OBUs), register with the ESP and communicate via Dedicated Short-Range Communications [27]. This setup enables mutual authentication between EVs and CSs, allowing session key formation without ESP involvement. The ESP periodically updates the secure database sent to CSs, ensuring efficient and secure operations within the AI-Enhanced Secure Protocol framework for energy management in Industrial Cyber-Physical Systems.

D. Threat Model

The proposed V2G protocol involves distinct communication phases. Initially, user registration with the Electric Service Provider (ESP) occurs over a secure channel. Subsequently, during the execution phase, all entities including Electric Vehicles (EVs) and Charging Stations (CSs) communicate over an insecure public channel, in accordance with the Dolev-Yao threat model [28], which anticipates potential adversary actions such as message interception, modification, or deletion.

Public network reliance exposes the V2G system to cyber threats like impersonation and man-in-the-middle attacks, compromising user privacy and system integrity. These vulnerabilities highlight the critical need for a robust authenticated key agreement protocol. This protocol should validate entity legitimacy and establish secure session keys, thereby fortifying the system against cyber threats, especially in Industrial Cyber-Physical Systems.

III. Proposed Protocol

In this section, we present the proposed lightweight authentication protocol for the Vehicle-to-Grid (V2G) communication environment. Consider a scenario where a user, denoted as EV_i, equipped with Internet connectivity, seeks to charge the battery at a charging station CS_i. It is imperative that EV_i and CS_i authenticate each other with the assistance of the Electric Service Provider (ESP). Upon successful mutual authentication, EV_i and CS_i establish a session key SK to secure their communications. The proposed protocol is structured into two phases: the registration phase and the authentication phase.

A. Registration Phase

Each EV must undergo a registration process with the Electric Service Provider (ESP), encompassing the following steps:

1) The user, denoted as EV, initiates the process by sending their credentials V_{id}, V_{pwd}, and β to the ESP via a secure channel. The EV generates a random number $r_v \in Z_p^*$, computes $PID_v = h(V_{id} \| r_v)$, and shares it with the ESP securely.
2) Upon receipt of the registration request, the ESP creates a new account for EV_i and updates its database. The ESP verifies the unique pseudo identity EV_{pid}, generates a secret key K_v, and a set of shadow identities $SID = \{sid_i, sid_j, \ldots, sid_n\}$. These shadow identities are used for loss of synchronization scenarios. The ESP sends $\{EV_{pid}, K_{vi}, SID\}$ and two challenges α_1 and α_2 to EV_i via a secure channel, and records $\{EV_{pid}, K_{vi}, SID\}$ in its database.
3) Upon receiving $\{EV_{pid}, K_{vi}, SID\}$ and the challenges α_1 and α_2 from the ESP, EV_i generates its secret key K_v. It then computes ${K_{vi}}^* = K_{vi} \oplus h(\beta || V_{pwd})$, where h is a cryptographic hash function and $\oplus$ denotes bitwise XOR. Finally, EV_i generates $\beta_2 = WPUF_{EV}(\alpha_2)$ and $\beta_1 = RPUF_{EV}(\alpha_1)$, and stores $\{PID_i, {K_{vi}}^*, SID, \alpha_1, \alpha_2, \beta_1, \beta_2\}$ in their tamper-proof memory present on the OBU for subsequent communication with the ESP, also sharing these values securely with the ESP for storage in its database.

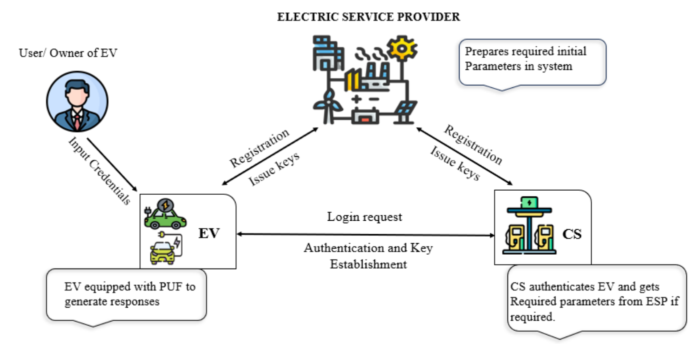

Fig. 2. Message Flow of Proposed Protocol

B. Login and Authentication Phase

To ensure communication security, each EV_i must undergo an authentication process before using a charging station CS_i. The authentication phase of the proposed protocol comprises the following steps:

EV_i starts by inputting their thumbprint β_i and password V_{pwd}. The device calculates $b_i = h(\beta_i)$ and $\partial_i' = h(b_i || V_{pwd})$ to validate the user's legitimacy. Upon successful validation, MD_i computes the key $k_v = k_v^* \oplus h(\beta_i || V_{pwd})$. The EV_i generates a nonce r_v and retrieves its location using Location Service (LS_v). The user then computes a key-hash response $L_1 = h(\text{EV}_{pid} || r_v || k_v || \text{EL})$, where $\text{EL} = \text{LS}_v \oplus h(k_v || r_v)$, and sends $\text{MSG}_1 = \{\text{EV}_{pid}, r_v, \text{EL}, L_1\}$ to the charging station CS_i.

Upon receiving MSG_1, the charging station CS_i generates a nonce r_{cs} and computes $L_2 = h(CS_{id} \| r_{cs} \| k_{cs} \| LS_{cs})$. It checks the authenticity of EV and sends $MSG_2 : \{MSG_1, CS_{id}, r_{cs}, LS_{cs}, L_2\}$ to the ESP.

The ESP retrieves EV_{pid} from its database, verifies the key-hash responses L_1 and L_2, decodes LS_v from EL, and validates against LS_{cs}. If successful, the ESP generates a session key SK and a new pseudo-identity EV_{pid}^{new}. It computes ${EV^{new*}}_{pid} = h(EV_{pid} \| K_v) \oplus EV_{pid}^{new}$, $SK_v = h(V_{id} \| K_v \| r_v) \oplus SK_{esp}$, $SK_{cs} = h(CS_{id} \| K_{cs} \| r_{cs}) \oplus SK_{esp}$, $L_3 = h(SK_{cs} \| K_{cs} \| r_e)$, and

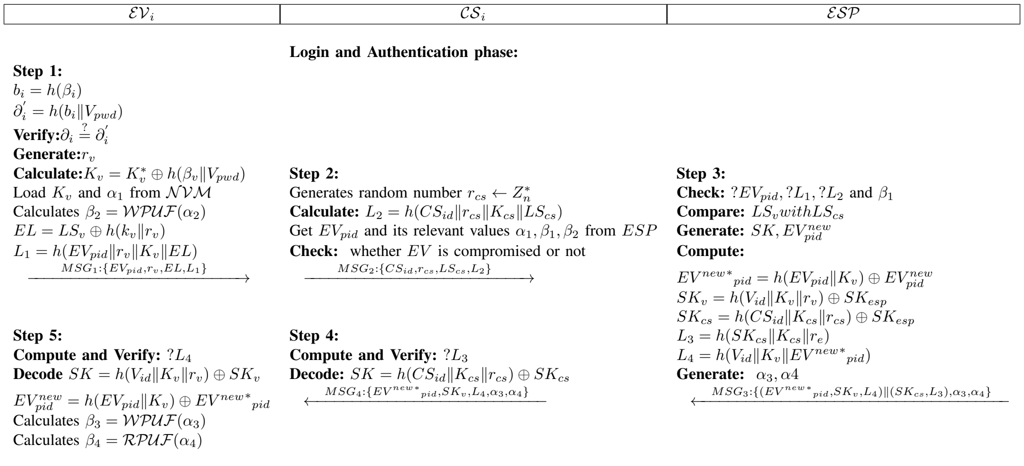

Fig. 3. Login and Authentication Phase of Proposed Protocol

$L_4 = h(V_{id} \| K_v \| EV^{new*}{}_{pid})$. The ESP sends $MSG_3 : \{(EV^{new*}{}_{pid}, SK_v, L_4) \| (SK_{cs}, L_3), \alpha_3, \alpha_4\}$ to CS_i.

CS_i receives MSG_3, validates L_3, and decodes SK by computing $SK = h(CS_{id} \| K_{cs} \| r_{cs}) \oplus SK_{cs}$. It sends $MSG_4 : \{EV^{new*}{}_{pid}, SK_v, L_4, \alpha_3, \alpha_4\}$ to EV_i.

EV_i receives MSG_4, verifies L_4, computes $SK = h(V_{id} \| K_v \| r_v) \oplus SK_v$, and calculates the new pseudo-identity EV_{pid}^{new} for subsequent communication using $EV_{pid}^{new} = h(EV_{pid} \| K_v) \oplus EV^{new*}{}_{pid}$. EV_i computes β_3 and β_4 using $\mathcal{WPUF}$ and $\mathcal{RPUF}$, respectively. If all checks pass, a session is established between EV and CS.

If any verification step fails, the protocol execution is terminated. To address synchronization loss, the following procedure is implemented:

If synchronization is lost, EV_i selects an unused shadow identity SID_n from $\{SID = sid_i, sid_j, \ldots, sid_n\}$ and sends it in MSG_1. Upon validation, the ESP generates a new pseudo-identity and securely transmits it in MSG_3 using K_v. After authentication, both EV_i and the ESP delete the used shadow identity. EV_i can use up to k shadow identities, where $k < n - 1$. Upon exhausting these identities, the user must request a reload. For a reload, EV_i sends a "Re-Load" message to the ESP, which generates new shadow identities and securely communicates them in MSG_3 using K_v.

Details of this phase and the associated mechanisms are depicted in Figure 3.

Operational considerations and advantages of the proposed protocol include repeated authentication for each transaction, ensuring robust security, maintaining location privacy by requiring anonymous authentication each time, and utilizing lightweight cryptographic primitives for reduced computational burden. Communication efficiency is significantly improved, as shown in Table IV, and the protocol allows a single account for multiple EVs, simplifying credential management.

In scenarios where two users, EV_i and EV_j, share a vehicle, the ESP generates two sets of security credentials: EV_{pidi}, K_{vi}, SID_i for EV_i and EV_{pidj}, K_{vj}, SID_j for EV_j, both linked to the same account. These credentials are securely stored by each user. When EV_i or EV_j uses the vehicle, they authenticate with their respective credentials. This shared usage model maintains flexibility and user-friendliness while ensuring secure authentication. The storage complexity at the ESP increases linearly with the number of users sharing the vehicle.

IV. Security Analysis

This section presents a rigorous proof of security for the proposed authentication scheme, confirming its robustness.

A. Formal Security Analysis

1) Definitions and Assumptions: Initially introduced by Bellare and Rogaway, the BR93-Model [29] provides a foundational framework for the security analysis of authentication and key exchange protocols. In our context, only the $\mathcal{ESP}$ has the capability to authenticate an $\mathcal{EV}$ directly, while the $\mathcal{CS}$ is responsible for relaying user authentication requests to the $\mathcal{ESP}$. It is presumed that the communication between the $\mathcal{CS}$ and $\mathcal{ESP}$ is secure, allowing them to be treated collectively as a single entity known as the $\mathcal{ESP}$.

2) Complexity Assumptions: The security of our proposed model is underpinned by the assumption that one-way hash functions employed are secure pseudorandom functions [30]. We detail the definitions related to pseudorandom functions and describe the game scenarios utilized for these security proofs.

Definition 1: Define f as a polynomial-time computable function. Let $\text{Adv}_H = |\Pr[H_f = 1] - \Pr[H_{f'} = 1]|$ signify the advantage that an algorithm H, managed by a polynomial-time adversary $\mathcal{ADV}$, has in distinguishing f from another function f'. Function f is deemed (n, q, ϵ)-secure as a pseudorandom function if no algorithm H exists that can distinguish f from f' with advantage $\text{Adv}_H \geq \epsilon$, given at most q oracle queries to f or a truly random function f', within at most n operations.

The game is structured as follows:

- *Initialization*: Challenger $\mathcal{C}$ interacts with $\mathcal{ADV}$, choosing a random bit $bt \in \{0, 1\}$ to set the function f_{bt}, where f_0 is a pseudorandom function, and f_1 is a truly random function.

- *Training Phase*: $\mathcal{ADV}$ sends up to q queries, $x_1, \ldots, x_q$, where each $x_i \in \{0,1\}^*$ is a binary string of arbitrary length. $\mathcal{C}$ responds with $f_{bt}(x_i)$ for $i = 1, \ldots, q$.
- *Guess*: $\mathcal{ADV}$ concludes by guessing a bit bt', attempting to identify bt. The game is won by $\mathcal{ADV}$ if $bt' = bt$, quantified by $\mathcal{ADVT}_{f_0,ADV} = |\Pr[bt' = bt] - 1/2|$.

Based on the assumption of the pseudorandom function, it is established that no adversary with probabilistic polynomial-time capabilities can achieve a non-negligible advantage in this game.

3) Security Model and Notations: The entities in the protocol are represented as oracles $\Pi_s^{A,B}$ and $\Pi_t^{B,A}$, indicating interaction in a session s or t where $A, B \in I$ and I is the set of participant identities.

Protocols: Our scheme implements a three-party authentication mechanism but reduces to a two-party system for practicality. Consequently, we outline the two-party authentication and key exchange protocol as follows.

Definition 2: A two-party authentication and key exchange protocol, P, is formally defined by an efficiently computable function Π with the following inputs:

- k: Security parameter length.
- A: Identity of the initiator.
- B: Identity of the intended partner.
- x: Secret information.
- $\mathcal{K}$: Previous conversations.
- r: Random coin flips by the initiator.

The output from $\Pi(k, A, B, x, \mathcal{K}, r)$ includes the next message m, a decision δ, and a private output α.

4) Adversary Model: An adversary $\mathcal{ADV}$ operates as a probabilistic polynomial-time Turing machine, capable of manipulating communications between A and B. This includes eavesdropping, message alteration, and session secret compromise. These behaviors can be represented through the following queries.

$Execute(\Pi_s^{A,B}, \Pi_t^{B,A})$: This query enables the $\mathcal{ADV}$ to simulate passive observation during a communication session between two protocol participants. When this query is activated, the $\mathcal{ADV}$ gains the ability to transparently witness the entire communication exchange between the entities $\Pi_s^{A,B}$ and $\Pi_t^{B,A}$, representing the $\mathcal{EV}$ and the $\mathcal{ESP}$, respectively.

$Send(\Pi_s^{A,B}, msg)$: This query exemplifies an active attack scenario, where the $\mathcal{ADV}$ is not merely an observer but an interactor. This query allows the $\mathcal{ADV}$ to actively participate in the communication process by injecting or altering messages. Specifically, when this query is executed, the adversary sends a crafted message msg to a participating entity $\Pi_s^{A,B}$. This entity processes the message as if it were received from its legitimate communication partner, and responds according to the protocol rules.

$Reveal(\Pi_s^{A,B})$: This query facilitates an analysis of the protocol's resilience against the exposure of sensitive session information. Within this framework, an $\mathcal{ADV}$ invokes the *Reveal* query to obtain the session key from an ongoing or completed session, represented by $\Pi_s^{A,B}$.

$Corrupt(\Pi_s^{A,B})$: This query models a significant security breach scenario within the protocol, where the $\mathcal{ADV}$ gains access to long-term secret keys or other critical state information maintained by a protocol participant.

$Test(\Pi_s^{A,B})$: This query is designed to evaluate the indistinguishability of session keys from random strings, a critical component in assessing the effectiveness of cryptographic protocols. During this query, once a session between $\Pi_s^{A,B}$ and $\Pi_t^{B,A}$ has successfully concluded and both parties have accepted a session key, the $\mathcal{ADV}$ can issue the *Test* query to one of the session oracles. The queried oracle then provides either the genuine session key or a random string, determined by a random coin flip. This query enables the $\mathcal{ADV}$ to attempt distinguishing the real session key from the random string, offering insights into the cryptographic strength of the session key generation process and the overall security of the protocol.

5) Security Definitions: Prior to introducing the concept of mutual authentication security, we will briefly analyze the definition of a matching conversation.

Definition 3: A protocol session is defined by (A, B, s, role). Two sessions have a matching conversation if their session identifiers are the same and they involve the same initiator and responder. Mutual authentication is defined as both parties accepting each other following a matching conversation, with a negligible probability that acceptance occurs without a matching conversation.

Definition 4: A protocol P is Mutual Authentication Secure (MA-secure) if acceptance by any oracle implies a matching conversation and vice versa.

Definition 5: Protocol P is Authentication Key Exchange Secure (AKE-secure) if it ensures mutual authentication and resists all known forms of attacks where $\mathcal{ADV}$ could distinguish a session key from a random string under the MA-security model.

B. Detailed Security Evaluation of the Proposed Protocol

This analysis details the robustness of the authentication scheme, hinged on the secure pseudorandom nature of hash functions employed within the protocol. Our framework, although conceptualized as a three-entity system involving the $\mathcal{EV}$, $\mathcal{CS}$, and $\mathcal{ESP}$, fundamentally operates as a two-party protocol in practical scenarios.

Lemma 1: If the hash function h is a (n_0, q_0, ϵ_0)-secure pseudorandom function with ϵ_0 being negligible, then the authentication protocol is assuredly $MA-Secure$.

Proof: Consider an adversary $\mathcal{ADV}$ that attempts to compromise the MA-Security of our protocol P. The probability of success by $\mathcal{ADV}$, denoted as $Success_P^{MA}$ ($\mathcal{ADV}$), combines the probabilities of impersonating a legitimate $\mathcal{EV}$ or the $\mathcal{ESP}$. The analysis bifurcates into two distinct impersonation attempts:

$Case1$ (Impersonating as $\mathcal{ESP}$) : Suppose $\mathcal{ADV}$ attempts to impersonate the $\mathcal{ESP}$ with a probability ϵ'. Within this setup, to validate authentication to an $\mathcal{EV}_i$ using $\Pi_s^{\mathcal{EV},\mathcal{ESP}}$, $\mathcal{ADV}$ needs to correctly generate the response $L_3 = h(SK_{cs}\|K_{cs}\|r_e)$. Here, the simulator $\mathcal{F}$ challenges $\mathcal{ADV}$ by simulating a game under the definitions of a secure pseudorandom function:

Initialization: $\mathcal{F}$ configures the hash function h_{bt} where $h_0 = h_{ki}$, and k_i is a long term secret key of length k-bit, switching between a pseudorandom function h_0 and a truly random function h_1 based on a random bit bt choosen by $\mathcal{C}$.

Training Phase: F mimics the roles of $\mathcal{EV}$ as $\Pi_s^{\mathcal{EV},\mathcal{ESP}}$ and $\mathcal{ESP}$ as $\Pi_s^{\mathcal{ESP},\mathcal{EV}}$, responding to $\mathcal{ADV}$'s authentication attempts by employing h_{bt} and monitoring the accuracy of $\mathcal{ADV}$'s responses by following queries:

- $Execute(\Pi_s^{\mathcal{EV},\mathcal{ESP}}, \Pi_t^{\mathcal{ESP},\mathcal{EV}})$: Utilizing the hash function h_b provided by challenger $\mathcal{C}$ as h_{ki} within the protocol dynamics, $\mathcal{F}$ generates random values for k_h and EV_{pid}^{new}. It then computes $EV^{new*}{}_{pid} = h(EV_{pid}\|K_v) \oplus EV_{pid}^{new}$, $SK_v = h(V_{id}\|K_v\|r_v) \oplus SK_{esp}$, and $L_4 = h(V_{id}\|K_v\|EV^{new*}{}_{pid})$, effectively simulating the roles of both the $\mathcal{EV}$ and the $\mathcal{ESP}$.
- $Send(\Pi_s^{\mathcal{EV},\mathcal{ESP}}, msg)$: In this step, $\mathcal{F}$ as the $\mathcal{EV}$ sends the protocol initiation message $MSG_1 : \{EV_{pid}, r_v, EL, L_1\}$ to simulate a normal protocol operation. The simulator then validates L_1 using h_b to verify the message's authenticity before proceeding to the next step.
- $Send(\Pi_s^{\mathcal{ESP},\mathcal{EV}}, msg)$: Upon receiving the message msg, $\mathcal{F}$ as the $\mathcal{ESP}$ computes $EV^{new*}{}_{pid} = h(EV_{pid}\|K_v) \oplus EV_{pid}^{new}$ and $L_4 = h(V_{id}\|K_v\|EV^{new*}{}_{pid})$ based on the received data and the earlier prepared values. The computed values help $\mathcal{F}$ to simulate a legitimate response from the $\mathcal{ESP}$ back to the $\mathcal{EV}$, encapsulated in the message $\{(EV^{new*}{}_{pid}, SK_v, L_4)\|(SK_{cs}, L_3), \alpha_3, \alpha_4\}$, maintaining the integrity and flow of the protocol simulation.
- $Challenge$: To instigate the protocol, $\mathcal{ADV}$ sends a request which triggers the simulated protocol execution. The request message sent by $\mathcal{ADV}$ is $MSG : \{EV_{pid}, r_v, EL, L_1\}$. Upon receipt, $\mathcal{F}$ evaluates L_1 for correctness using h_{bt}, proceeding to simulate the correct protocol behavior as expected in a genuine interaction scenario. Subsequently, $\mathcal{ADV}$ generates and sends the expected authentication response L_4, attempting to complete the authentication phase successfully. The simulator $\mathcal{F}$ then issues a final verification query $x^* = h(\mathrm{SK}_v \parallel K_i)$ to h_{bt} and receives the output $L_3 = h(SK_{cs}\|K_{cs}\|r_e)$, concluding the simulation and assessment phase.

Guess Phase:

During the simulation's final phase, simulator F assesses its ability to discern the function h_b used by adversary A, based on the outcomes of the authentication tests. The effectiveness of F in this regard is ascertained through the following analytical steps:

- Decision Making: If the authentication response calculated by $\mathcal{F}$, denoted as L_3, matches the response L_4 provided by $\mathcal{ADV}$, then $\mathcal{F}$ confirms the test as successful by outputting $bt' = 0$. If they do not match, $\mathcal{F}$ randomly chooses to output either 0 or 1.
- Probability Analysis: The capability of $\mathcal{F}$ to correctly identify whether $\mathcal{ADV}$ is using a pseudorandom function (h_0) or a truly random function (h_1) is evaluated under two experimental conditions:
  - Real Experiment: Here, $bt = 0$, meaning $h_{bt} = h_0$ is pseudorandom. If $\mathcal{ADV}$ successfully mimics legitimate authentication behavior, $\mathcal{F}$ outputs $bt' = 0$ with a probability of ϵ'. Conversely, if $\mathcal{ADV}$ errs, $\mathcal{F}$ randomly guesses, resulting in a correct guess with a probability of $(1-\epsilon')/2$.
  - Random Experiment: Here, $bt = 1$, meaning $h_{bt} = h_1$ is truly random. $\mathcal{ADV}$ has no insight into h_1, hence can only guess correctly with a probability corresponding to a random guess, 2^{-k}, leading to $\mathcal{F}$ accurately guessing bt' with a probability of $(1 - 2^{-k})/2$.
- Combined Outcome: The total probability of $\mathcal{F}$ accurately predicting b is a cumulative measure from both scenarios:

$$Pr[bt' = bt] = \left(\epsilon' + \frac{1-\epsilon'}{2}\right)\cdot\frac{1}{2} + \left(\frac{1-2^{-k}}{2}\right)\cdot\frac{1}{2}$$
$$= \frac{1}{2} + \frac{\epsilon'}{4} - 2^{-(k+2)}.$$

- Implication: The derived probabilities indicate that if $\mathcal{F}$ discerns b with a significantly high accuracy exceeding $\frac{1}{2}$ by a margin of $\epsilon'/4 - 2^{-(k+2)}$, it suggests that ϵ' is constrained by the bounds of $4\epsilon_0 + 2^{-k}$, confirming the strength of the pseudorandom function used in the protocol.

Conclusive Guess: As the final step in the simulation, $\mathcal{F}$ outputs a guess bit bt', consolidating its assessment of whether $\mathcal{ADV}$ managed to convincingly impersonate legitimate protocol interactions using either a pseudorandom or truly random hash function. The outcome of this guess ultimately validates the security efficacy of the authentication mechanism employed within the protocol.

$Case2$ (Impersonating as $\mathcal{EV}$): In this scenario, $\mathcal{ADV}$'s success hinges on fabricating $\mathcal{EV}$-specific credentials and session keys. Similar to the $\mathcal{ESP}$ impersonation case, $\mathcal{F}$ leverages h_{bt} to evaluate $\mathcal{ADV}$'s proficiency in generating valid session communications under the guise of a legitimate $\mathcal{EV}$, adhering to the predefined game rules.

Lemma 2: Post validation of MA-Security in Lemma 1, the protocol's AKE-Security is asserted if h maintains its integrity as a secure pseudorandom function.

Proof: Building on Lemma 1, which confirms that protocol $\mathcal{P}$ is MA-Secure, we now assess the protocol's resilience against adversaries capable of breaching AKE-Security. Consider an adversary $\mathcal{ADV}$ that can challenge the AKE-Security of $\mathcal{P}$ with a non-negligible advantage, denoted as ϵ.

Simulation Setup by $\mathcal{F}$: $\mathcal{F}$ constructs a simulation to test the capabilities of $\mathcal{ADV}$ under the assumption that pseudorandom functions can be compromised. This testing follows the specifications outlined in Definition 3, with $\mathcal{C}$ providing the necessary setup.

- Initialization: The challenger $\mathcal{C}$ selects a random bit $bt \in \{0,1\}$, configuring h_{bt} as either a pseudorandom function $h_0 = h_{ki}$ or a truly random function h_1.
- Training Phase: $\mathcal{F}$ engages with $\mathcal{ADV}$ by simulating both $\mathcal{EV}$ and $\mathcal{ESP}$ roles, using h_{bt} to respond to executions

and data requests in the protocol. This simulation mirrors the scenarios explored in Lemma 1.

- Testing Phase: During the testing:
  - If the session key KEY of $\Pi_s^{\mathcal{EV},\mathcal{ESP}}$ is generated, $\mathcal{F}$ randomly selects $z \in \{0,1\}$, returning the actual session key if $z = 0$, or a random string if $z = 1$. If KEY is not generated, $\mathcal{F}$ returns $\perp$, representing an invalid or undefined outcome.
- Challenge and Response: Following the execution and send queries, $\mathcal{ADV}$ is prompted to test the authenticity of the session key through a test query to $\mathcal{F}$. $\mathcal{ADV}$'s response indicates whether it perceives the key as legitimate or fabricated.
- Final Guess by $\mathcal{F}$: After $\mathcal{ADV}$ submits its guess, $\mathcal{F}$ evaluates whether $\mathcal{ADV}$'s perception aligns with the actual scenario (real vs. simulated). $\mathcal{F}$ concludes this phase by outputting a guess bt', determining whether bt' matches bt based on $\mathcal{ADV}$'s responses and the setup of h_{bt}.

Probabilistic Analysis: The probability that $\mathcal{F}$ correctly identifies bt as bt', denoted as $Pr[bt = bt']$, combines the outcomes from both the real and the random experiments:

$$\begin{aligned} Pr[bt = bt'] &= Pr[bt = bt', bt = 0] + Pr[bt = bt', bt = 1] \\ &= \left(\epsilon + \frac{1}{2}\right) \times \frac{1}{2} + \frac{1}{4} \\ &= \frac{1}{2} + \frac{\epsilon}{2}. \end{aligned}$$

This outcome implies that if ϵ_0, representing the minimum detectable effect size, is significant, a contradiction is evident, proving that the adversary's advantage, ϵ, is insufficient to compromise the protocol meaningfully.

Conclusion: Hence, it is established that the Advantage of $\mathcal{ADV}$ under the AKE-Security framework, $\mathcal{ADVT}_{AKE}^{P}(\mathcal{ADV})$, is negligible, reinforcing the robustness of protocol $\mathcal{P}$ against all polynomial-time adversaries. Thus, $\mathcal{P}$ is validated as AKE-Secure.

C. Informal Security Analysis

This subsection informally examines how our authentication protocol ensures key security features and meets specific requirements for the V2G communication environment.

1) AI Integration for Anomaly Detection: In developing our AI-enhanced secure protocol for Industrial Cyber-Physical Systems, we incorporated LSTM networks for Anomaly Detection. Using the KDD Cup 1999 dataset, our model trained on historical operational patterns and security incidents. This improved our protocol's ability to promptly identify and address security threats, enhancing the reliability and safety of Vehicle-to-Grid communications. Our implementation demonstrates our commitment to using advanced AI technologies to strengthen security measures against cyber threats.

2) Impersonation and Forgery Attack Prevention: Our protocol prevents impersonation and forgery attacks in V2G communications, ensuring integrity and authenticity:

User Impersonation: Without the user's unique β and V_{pwd}, an adversary cannot compute K_v, EL, or a valid L_1, making impersonation impossible.

Service Provider Impersonation: Lacking secret keys K_v and K_{cs}, an adversary cannot generate valid key-hash responses L_3 and L_4.

Location Identity Forgery: A false LS_{cs} is countered by the ESP, which validates LS_v from EL against LS_{cs}. Discrepancies terminate the protocol and flag CS_i.

User Device Loss or Theft: Multi-factor security involving β and V_{pwd} prevents unauthorized protocol execution by an adversary.

These measures effectively mitigate impersonation and forgery risks, strengthening AI-based V2G communication security.

3) Privacy and Identity Intractability: Our protocol ensures user privacy and identity intractability in V2G communication, protecting against eavesdropping:

Use of Pseudo Identity: Each session uses a unique pseudo identity EV_{pid}, preventing unauthorized tracking and linking of actions over time.

Handling Loss of Synchronization: Users switch to an unused shadow identity SID_n if synchronization is lost. Used shadow identities are discarded, enhancing privacy.

Privacy Against Eavesdropping: Constantly changing pseudo-identities and using shadow identities prevent eavesdroppers from correlating sessions to a specific user.

These measures preserve user confidentiality and privacy in V2G environments.

4) Protection Against Stolen/Compromised Device: Our proposed protocol incorporates measures to address scenarios where an attacker gains control over a user's vehicle and alters their credentials:

- Immediate Notification: In case of a security breach, the legitimate user is urged to inform the Electric Service Provider (ESP) immediately. Prompt reporting is crucial to prevent further unauthorized access or misuse.
- Account Blocking: Upon notification, the ESP takes swift action to block the user's account, halting any further transactions by the adversary with the compromised device.
- Limit on Transactions: As an additional layer of security, the ESP may set a limit on weekly or monthly charging/discharging activities for users. This cap ensures that, even if a device is compromised, the extent of unauthorized usage is restricted.

These strategies collectively strengthen the protocol's defence against scenarios involving stolen or compromised devices, enhancing the security framework of the protocol.

5) Protection Against Physical Attacks and Invasive Assaults: Our protocol implements stringent measures to safeguard against physical and invasive attacks on secret credentials stored in device memory, especially within the $WPUF$ and $RPUF$ contexts.

For the WPUF, if an adversary $\mathcal{ADV}$ tries to extract secrets like K_v and α_2 from WPUF's RAM, the WPUF's altered behavior disrupts its output $\beta_2 = \mathcal{WPUF}(\alpha_2)$. This anomaly signals to the ESP that a breach is being attempted, allowing for timely detection and response.

For the RPUF, if an $\mathcal{ADV}$ targets RPUF's memory, any change in RPUF behavior is detected by the CS. This enables the CS to recognize and react appropriately to such intrusion attempts.

These protective mechanisms ensure robust defense against physical and invasive attacks, preserving the integrity of the system's hardware components and enhancing vehicular security.

6) Defense Against Machine Learning Attacks: Our protocol counters machine learning attacks using a reconfigurable $\mathcal{RPUF}$. By adjusting the refresh pause interval, the $\mathcal{RPUF}$ behavior is modified, introducing variability and thwarting predictive modeling. Even if an $\mathcal{ADV}$ obtains several Challenge-Response Pairs, creating a soft model of the $\mathcal{RPUF}$ remains challenging. The reconfigurable nature of the $\mathcal{RPUF}$ changes its performance after each session, complicating predictive modeling attempts. The unpredictable $\mathcal{RPUF}$ responses render machine learning attacks ineffective, as maintaining an accurate model becomes infeasible. This strategy makes our protocol resilient against machine learning attacks, enhancing overall security.

V. Performance Evaluation

This section compares the proposed protocol with existing V2G authentication protocols, focusing on security features, communication, and computing costs. The names of security features that are featured in Table III are EV impersonation, CS impersonation, Man In Middle (MIM), Distributed Denial of Service (DDOS), Privileged insider, Replay, User anonymity, Forward and Backward secrecy, Desynchronization, Physical and Machine learning attacks.

A. Security Feature Analysis

Our comparative analysis evaluates the security strength of our proposed protocol against recognized attacks, benchmarked against protocols in [4], [5], [7]–[11]. "Y" indicates support for a security property or attack resilience, while "N" signifies absence or attack susceptibility, as summarized in Table III.

Forward secrecy is ensured by our protocol and those in [4], [5], [7]–[10], unlike [11]. Physical attack susceptibility is noted in all protocols except [10], [11]. User anonymity is provided by protocols in [4], [5], [9], [10], but not by [8]. The protocol in [5] is vulnerable to man-in-the-middle attacks. All protocols except [10], [11] and our proposed protocol are vulnerable to machine learning attacks.

B. Communication Cost Analysis

The communication cost of our proposed protocol, measured in bytes, focuses on data transmitted during mutual authentication. We use SHA-256 and AES encryption with 320-bit ECC-based point multiplication, 32-bit timestamp, 64-bit identity, 64-bit random number, and 128-bit PUF responses.

During login and authentication, the EV sends MSG_1 : $\{EV_{pid}, r_v, EL, L_1\}$ (256 bits) to the CS. The CS responds with MSG_2 : $\{CS_{id}, r_{cs}, LS_{cs}, L_2\}$ (512 bits). The EV's MSG_3 and MSG_4 are 320 bits and 576 bits, respectively. The total communication is 1728 bits or 216 bytes.

Our protocol reduces communication costs by 22.80%, 28.95%, 35.71%, 48.32%, and 58.14% compared to [4], [5], [9], and [7], respectively. There is a slight increase in overhead compared to [8], [10], and [11], which we accept for significant security improvements, detailed in Table IV.

TABLE III
Security Features Comparison

Protocols	F1	F2	F3	F4	F5	F6	F7	F8	F9	F10	F11
[4]	N	N	N	Y	Y	N	Y	Y	N	N	N
[5]	Y	Y	N	N	Y	Y	Y	Y	N	N	N
[7]	Y	Y	Y	Y	Y	N	N	Y	N	N	N
[8]	Y	Y	Y	N	N	Y	Y	Y	N	Y	N
[9]	Y	N	N	Y	Y	Y	Y	N	Y	N	N
[10]	Y	Y	Y	Y	Y	Y	Y	Y	N	Y	Y
[11]	Y	Y	N	Y	N	Y	Y	N	Y	Y	Y
Proposed Protocol	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y

TABLE IV
Computation & Communication Cost Analysis

Reference	$User$ Device	Results (in ms)	Communication Cost (in bytes)
[4]	$4T_{pm} + T_{me} + 5T_h$	2802	280
[5]	$3T_{pm} + T_{me} + 6T_h$	2274	304
[7]	$T_{fe} + 9T_h$	218	516
[8]	$T_{fe} + 2T_{PUF} + 7T_h$	110.9	220
[9]	$T_{fe} + T_{PUF} + 3T_h$	185.4	312
[10]	$T_{WPUF} + 2T_{RPUF} + 7T_h$	63.6	208
[11]	$T_{WPUF} + 2_{RPUF} + 7T_h$	63.6	208
Proposed	$T_{WPUF} + 2T_{RPUF} + 8T_h$	65.2	216

C. Computational Cost Analysis

The computational cost of our protocol is measured in milliseconds (ms), focusing on cryptographic operations for communication. Reduced computation time facilitates faster interactions between entities. We analyze computational expenses at the user device (EV) for Random PUF (RPUF) response generation T_{RPUF}, modular exponentiation T_{me}, hash functions T_h, ECC point multiplication T_{pm}, PUF response generation T_{PUF}, symmetric encryption T_{Senc}, and WPUF response generation T_{WPUF}.

Measurements were taken using a Redmi Note 11 as EV and an HP Probook G7 as CS and ESP, utilizing the Bouncy Castle Library. Public key-based protocols [4], [5] incur higher costs, while protocols [10], [11] address physical and machine learning attacks.

Our protocol shows significant cost reductions compared to [4], [5], [7], [8], and [9], achieving efficiencies of 97.67%, 97.13%, 70.09%, 6.84%, 68.02%, and 63.62%, respectively.

TABLE V
Execution Time of Cryptographic Operations

Cryptographic Operation	User Device	ESP/CS
T_{pm}	5.09 ms	2.4 ms
T_m	20.23 ms	12.4 ms
T_h	0.0186 ms	0.013 ms
T_{Senc}	0.053 ms	0.039 ms
T_{PUF}	3.81 ms	2.57 ms
T_{WPUF}	2.221 ms	1.79 ms
T_{RPUF}	3.321 ms	2.34 ms

However, there is a 2.52% increase in computation overhead compared to [10] and [11]. Details are in Table IV.

VI. Conclusion

The essence of secure and efficient key exchange in the V2G system is central to this manuscript, where we introduced an AI-Enhanced Secure Protocol for V2G communication in Industrial Cyber-Physical Systems. Our protocol incorporates lightweight cryptographic elements, notably non-collision one-way hash functions, boosting the protocol's efficacy and practicability. Rigorous theoretical analysis and simulation tools have quantified our protocol's performance, demonstrating resilience against various security attacks while ensuring high computational and communication efficiency. Comparative studies reveal our method's superior security attributes and operational effectiveness, making it an apt solution for secure key exchange in the evolving V2G landscape. This aligns with our goal of optimizing V2G dynamics, addressing both the imperative of security and practical efficiency considerations in real-world applications.

References

[1] Abbas Mehrabi and Kiseon Kim. Low-complexity charging/discharging scheduling for electric vehicles at home and common lots for smart households prosumers. *IEEE Transactions on Consumer Electronics*, 64(3):348–355, 2018.

[2] Vikas Hassija, Vinay Chamola, Sahil Garg, Dara Nanda Gopala Krishna, Georges Kaddoum, and Dushantha Nalin K Jayakody. A blockchain-based framework for lightweight data sharing and energy trading in v2g network. *IEEE Transactions on Vehicular Technology*, 69(6):5799–5812, 2020.

[3] Ponnuru Raveendra Babu, Basker Palaniswamy, Alavalapati Goutham Reddy, Vanga Odelu, and Hyun Sung Kim. A survey on security challenges and protocols of electric vehicle dynamic charging system. *Security and Privacy*, 5(3):e210, 2022.

[4] Jia-Lun Tsai and Nai-Wei Lo. Secure anonymous key distribution scheme for smart grid. *IEEE transactions on smart grid*, 7(2):906–914, 2015.

[5] Vanga Odelu, Ashok Kumar Das, Mohammad Wazid, and Mauro Conti. Provably secure authenticated key agreement scheme for smart grid. *IEEE Transactions on Smart Grid*, 9(3):1900–1910, 2016.

[6] Akber Ali Khan, Vinod Kumar, Musheer Ahmad, Brij B Gupta, Musheer Ahmad, and Ahmed A Abd El-Latif. A secure and efficient key agreement framework for critical energy infrastructure using mobile device. *Telecommunication Systems*, 78:539–557, 2021.

[7] Azeem Irshad, Muhammad Usman, Shehzad Ashraf Chaudhry, Husnain Naqvi, and Muhammad Shafiq. A provably secure and efficient authenticated key agreement scheme for energy internet-based vehicle-to-grid technology framework. *IEEE Transactions on Industry Applications*, 56(4):4425–4435, 2020.

[8] Mingping Qi and Jianhua Chen. Two-pass privacy preserving authenticated key agreement scheme for smart grid. *IEEE Systems Journal*, 15(3):3201–3207, 2020.

[9] Masoud Kaveh and Mohamad Reza Mosavi. A lightweight mutual authentication for smart grid neighborhood area network communications based on physically unclonable function. *IEEE Systems Journal*, 14(3):4535–4544, 2020.

[10] Prosanta Gope and Biplab Sikdar. A privacy-aware reconfigurable authenticated key exchange scheme for secure communication in smart grids. *IEEE Transactions on Smart Grid*, 12(6):5335–5348, 2021.

[11] Prosanta Gope, Owen Millwood, and Biplab Sikdar. A scalable protocol level approach to prevent machine learning attacks on physically unclonable function based authentication mechanisms for internet of medical things. *IEEE Transactions on Industrial Informatics*, 18(3):1971–1980, 2021.

[12] Willett Kempton and Jasna Tomić. Vehicle-to-grid power implementation: From stabilizing the grid to supporting large-scale renewable energy. *Journal of power sources*, 144(1):280–294, 2005.

[13] Sekyung Han, Soohee Han, and Kaoru Sezaki. Development of an optimal vehicle-to-grid aggregator for frequency regulation. *IEEE Transactions on smart grid*, 1(1):65–72, 2010.

[14] Fabian Kennel, Daniel Görges, and Steven Liu. Energy management for smart grids with electric vehicles based on hierarchical mpc. *IEEE Transactions on industrial informatics*, 9(3):1528–1537, 2012.

[15] Amin Mohammadali, Mohammad Sayad Haghighi, Mohammad Hesam Tadayon, and Alireza Mohammadi-Nodooshan. A novel identity-based key establishment method for advanced metering infrastructure in smart grid. *IEEE Transactions on Smart Grid*, 9(4):2834–2842, 2016.

[16] Hasen Nicanfar and Victor CM Leung. Multilayer consensus ecc-based password authenticated key-exchange (mcepak) protocol for smart grid system. *IEEE Transactions on Smart Grid*, 4(1):253–264, 2013.

[17] Dapeng Wu and Chi Zhou. Fault-tolerant and scalable key management for smart grid. *IEEE Transactions on Smart Grid*, 2(2):375–381, 2011.

[18] Yu Zhan, Liguo Zhou, Baocang Wang, Pu Duan, and Benyu Zhang. Efficient function queryable and privacy preserving data aggregation scheme in smart grid. *IEEE Transactions on Parallel and Distributed Systems*, 33(12):3430–3441, 2022.

[19] Amir Masoud Aminian Modarres and Ghazaleh Sarbishaei. An improved lightweight two-factor authentication protocol for iot applications. *IEEE Transactions on Industrial Informatics*, 2023.

[20] Seyed Ahmad Soleymani, Shidrokh Goudarzi, Mohammad Hossein Anisi, Haitham Cruickshank, Anish Jindal, and Nazri Kama. Truth: Trust and authentication scheme in 5g-iiot. *IEEE Transactions on Industrial Informatics*, 19(1):880–889, 2022.

[21] Chien-Ming Chen, Lili Chen, Yanyu Huang, Sachin Kumar, and Jimmy Ming-Tai Wu. Lightweight authentication protocol in edge-based smart grid environment. *EURASIP Journal on Wireless Communications and Networking*, 2021:1–18, 2021.

[22] Shafiq Ahmed, Salman Shamshad, Zahid Ghaffar, Khalid Mahmood, Neeraj Kumar, Reza M Parizi, and Kim-Kwang Raymond Choo. Signcryption based authenticated and key exchange protocol for ei-based v2g environment. *IEEE Transactions on Smart Grid*, 12(6):5290–5298, 2021.

[23] Salman Shamshad, Khalid Mahmood, Usman Shamshad, Ibrar Hussain, Shafiq Hussain, and Ashok Kumar Das. A provably secure and lightweight access control protocol for ei-based vehicle to grid environment. *IEEE Internet of Things Journal*, 2023.

[24] Blaise Gassend, Dwaine Clarke, Marten Van Dijk, and Srinivas Devadas. Silicon physical random functions. In *Proceedings of the 9th ACM Conference on Computer and Communications Security*, pages 148–160, 2002.

[25] Thomas McGrath, Ibrahim E Bagci, Zhiming M Wang, Utz Roedig, and Robert J Young. A puf taxonomy. *Applied physics reviews*, 6(1), 2019.

[26] Ulrich Rührmair, Frank Sehnke, Jan Sölter, Gideon Dror, Srinivas Devadas, and Jürgen Schmidhuber. Modeling attacks on physical unclonable functions. In *Proceedings of the 17th ACM conference on Computer and communications security*, pages 237–249, 2010.

[27] John B Kenney. Dedicated short-range communications (dsrc) standards in the united states. *Proceedings of the IEEE*, 99(7):1162–1182, 2011.

[28] Danny Dolev and Andrew Yao. On the security of public key protocols. *IEEE Transactions on information theory*, 29(2):198–208, 1983.

[29] Mihir Bellare and Phillip Rogaway. Entity authentication and key distribution. In *Annual international cryptology conference*, pages 232–249. Springer, 1993.

[30] B. Schneier. *Applied Cryptography, Second Edition*, pages 675–741. John Wiley & Sons, Ltd, 2015.