big data and cognitive computing

MDPI

Article

ECQV-Based Lightweight Revocable Authentication Protocol for Electric Vehicle Charging

Abdullah M. Almuhaideb [1,*] and Sammar S. Algothami [2]

[1] SAUDI ARAMCO Cybersecurity Chair, Department of Networks and Communications, College of Computer Science and Information Technology, Imam Abdulrahman Bin Faisal University, P.O. Box 1982, Dammam 31441, Saudi Arabia

[2] Department of Computer Science, College of Computer Science and Information Technology, Imam Abdulrahman Bin Faisal University, P.O. Box 1982, Dammam 31441, Saudi Arabia

* Correspondence: amalmuhaideb@iau.edu.sa

Abstract: In the near future, using electric vehicles will almost certainly be required for the sustainability of nature and our planet. The most significant challenge that users are concerned about is the availability of electric vehicle charging stations. Therefore, to maximize the availability of electric vehicle charging stations, we suggest taking benefit from individual sellers who produce renewable energy from their homes or electric vehicle owners who have charging piles installed in their homes. However, energy services that are rapidly being offered by these businesses do not have a trust connection developed with the consumers and stakeholders in these new systems. Exchange of data related to electric vehicles and energy aggregators can be used to identify users' behavior and compromise their privacy. Consequently, it is necessary to set up a charging system that will guarantee privacy and security. Several electric vehicle charging systems have been proposed to provide security and privacy preservation. However, ensuring anonymity alone is not enough to guarantee protection from reconstructing the victim vehicle's route by the tracking adversary, even if the exchanged messages are completely anonymous. Furthermore, anonymity should not be absolute in order to protect the system and function as necessary by all entities. In this research, we propose an effective, secure, and privacy-preserving authentication method based on the Elliptic Curve Qu–Vanstone for an electric vehicle charging system. The proposed scheme provides all the necessary requirements and a reauthentication protocol to minimize the overhead of subsequent authentication processes. To create credentials and validate electric vehicles and energy aggregators, the scheme makes use of the Elliptic Curve Qu–Vanstone implicit certificate mechanism. The new protocols give EVs security and privacy while cutting computational time by 95% thanks to reauthentication, as demonstrated by the performance comparison with earlier works.

Keywords: electric vehicle (EV); charging system; authentication; Elliptic Curve Qu–Vanstone (ECQV); implicit certificate; privacy-preserving; un-linkability; anonymity; reauthentication

check for updates

Citation: Almuhaideb, A.M.; Algothami, S.S. ECQV-Based Lightweight Revocable Authentication Protocol for Electric Vehicle Charging. *Big Data Cogn. Comput.* **2022**, *6*, 102. https://doi.org/10.3390/bdcc6040102

Academic Editor: Min Chen

Received: 29 July 2022
Accepted: 20 September 2022
Published: 27 September 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.

1. Introduction

Transportation produces over 30% of global greenhouse gas (GHG) emissions, which has a big impact on air quality [1]. With environmental concerns and a reduction in fossil fuel consumption, countries are increasingly promoting clean renewable energy alternatives to fossil energy. The electric vehicle (EV) is a good choice for tackling the energy crisis and climate change because it is reasonably priced and emission-free. EVs have recently attracted a lot of attention as a way to cut fuel use and GHG emissions and increase energy efficiency [2].

For three primary reasons, the growth of EVs is expected to continue, even with a higher rate of adoption in the coming years [3]:

1. Clean-fuel vehicles and initiatives to reduce carbon emissions should be encouraged: Saudi Arabia, the world's leading oil producer, declared that at least 30% of the cars in its capital city will be electric by 2030. Similarly, China seeks 25% of all new cars to be electrified by 2025. The UK attempts to stop producing and selling fossil-fuel vehicles by 2030 [4].
2. Resolve uncertainty for EV drivers: EV drivers still face uncertainty, even though more people are adopting the technology. As reported by the climate group EV100's members, the most significant challenge that users are concerned about is the availability of EV charging stations (e.g., charging station location, EV parking space, and charge cost) [5,6]. As a result, charging stations need to be strategically positioned and used efficiently as the demand for EVs increases [7]. Because work was disrupted in major areas due to the COVID-19 pandemic, the installation of publicly available chargers increased by 45%, a slower rate than the 85% seen in 2019. The company also cited a persistent barrier as the lack of suitable vehicle types. The cost of buying an electric vehicle remains a significant barrier [5];
3. Make EV charging a smooth experience: Remote control using smartphone applications is one of the features of smart EV charging. This feature makes EV charging faster as well as easier to use and, hence, more accessible to a wider variety of clients [3].

Advancements in distributed renewable production, storage systems, and EVs are causing evolving energy systems to become more decentralized. Energy services are rapidly being offered by businesses (such as individual sellers who produce renewable energy from their homes or EV owners who have charging piles installed in their homes) that have not developed trust connections with the consumers and stakeholders in these new systems [8]. The quality of power distributed by the grid is impacted by unregulated electric vehicle charging systems, which results in significant load changes in the electrical grid. As a result, existing energy systems experience severe negative effects, such as higher load peaks, degradation in power quality, and higher consumption of energy [9].

While an EV is connected to an electric vehicle charging station (EVCS), the energy aggregator (EAG) and EV continually exchange information. The EAG functions as an information collector for EVs and controls the charging of the EVs. The EV reports confidential details, such as the EV's identity (ID), battery status, consumed energy, and geographical location, to potentially untrustworthy charging entities [10]. The EV can share its distance from the service provider to hide the vehicle's exact location from adversaries. However, it is feasible for adversaries to determine the EV's precise location when its distance is disclosed [11]. Additional critical information, including the EV's identification information, its owners, and its travel behavior, can be inferred from the given information [12]. The information shared can be subject to numerous forms of attacks, since the EV and EAG connect using the internet, Wi-Fi, Bluetooth, dedicated short-range communications (DSRC), etc. These attacks may lead to inconsistent battery charging, poor EVCS operation, money theft, incorrect payment transaction outcomes, and more. The primary attacks that could happen are denial-of-service (DoS) attacks, replay attacks, impersonation attacks, and man-in-the-middle (MITM) attacks [10].

1.1. Problem Statement

One of the biggest hurdles to the widespread adoption of EVs is EV charging. Legal authorities are responsible for issuing EV credentials, which allow vehicle identification and authentication. Because these credentials hold EVs' genuine identifiers, they can be used to track vehicles. Furthermore, certain data in the electrical transactions must be hidden to prevent personal information (e.g., EV identity, battery charge status, geographical location, payment information, etc.) from being leaked. Furthermore, if a vehicle is not validated, an adversary vehicle can easily imitate an authorized vehicle to broadcast false information. This is because trading data can be utilized to analyze individuals' behaviors and invade their privacy. For instance, EV charging schedules can reveal when an owner remains at home or outside, allowing potential criminals to attempt robbery [13]. Furthermore, the majority of deployed EV charging stations lack physical security and are seldom supervised; an adversary could cause damage to it or install malware in them. Such malware could be utilized to steal energy, obtain users' data (such as ID card number to impersonate their identity for a transaction), or disrupt EV charging by causing a denial of service [14]. Additionally, in a wireless sensor network (WSN), a passive adversary may secretly intercept messages and employ traffic analysis methods to deduce details about the structure of the network topology and the profiles of network entities [15]. As a result, how to protect end-user privacy while dealing with electrical transaction data becomes a significant privacy-preserving challenge for researchers throughout EV charging and discharging.

An Elliptic Curve Qu–Vanstone (ECQV)-based EV charging system may be created to address these issues and provide security and straightforward authentication to EV clients. The ECQV implicit certificate is necessary for enabling mutual authentication, key establishment, and secret key exchange for Internet of Things (IoT) devices. There have been several EV charging solutions suggested to offer security and privacy preservation. However, the terms privacy and anonymity are sometimes used interchangeably. Privacy is much broader and refers to all aspects of maintaining user privacy, whereas anonymity is focused on maintaining user identity confidentiality. These systems tend to lack in achieving a balance between the need for privacy (trade traceability to achieve anonymity, etc.) and security considerations.

1.2. Paper Motivation and Contribution

In an effort to meet the requirements for preserving privacy and security mentioned previously, these are the contributions made by this paper:

1. Present an ECQV-based authentication solution that is more effective at preserving privacy and providing secure authentication for electric vehicle charging stations;
2. Use Burrows–Abadi–Needham (BAN) logic and the AVISPA simulation tool to conduct a formal security study to demonstrate that the proposed scheme is secure against numerous attacks. In addition, we perform an informal security analysis to show the proposed protocol's security;
3. Compare the computational costs with other related work, to illustrate that the proposed techniques will perform better.

1.3. Paper Organization

Following is the format for the remaining portion of the paper: Section 2 covers the preliminary material. In Section 3, we show an analysis of the literature that is relevant to the proposed protocol. In Section 4, the proposed scheme is presented. The formal and informal security analyses are discussed in Section 5. In Section 6, security, functional, and computational aspects are compared with those of related schemes. Finally, Section 7 is the conclusion.

2. Preliminaries

The criteria for EV charging's authentication solution are covered in this section. Additionally, the concept of an elliptic curve Qu–Vanstone (ECQV) implicit certificate is introduced in this section.

2.1. Solution Requirements

Authentication that protects the privacy and the system's ability to thwart both active and passive attacks (against system entities) are necessary for the feasibility and acceptability of EV charging systems. These key security requirements must be met before a charging system may be used. However, the current approaches have flaws that raise several questions about EV charging systems. As a result, the proposed scheme needs to fulfill the following criteria:

1. Mutual authentication: The system must allow the parties to confirm one another's identities and guarantee that communication is based on trust. To verify the EAG's identification and registration with the trusted charging system operator (OP), the EV must authenticate the EAG. The EAG will verify the EV's registration with the OP concurrently. The OP issues certificates for authentication, consequently reducing the likelihood of a masquerade attack [16];
2. Anonymity: Anonymity is the capability to evade being recognized within a group of subjects. The EV's true identity should not be revealed to the EAG while it is charging [17]. Un-traceability is the ability to keep the activities of a subject un-traceable. Eavesdroppers cannot guess or trace the EV's activities [16];
3. Un-linkability: Un-linkability is where the attacker cannot tell whether two actions are related. EVs during various charging sessions should not be linkable [17];
4. Traceability: This characteristic guarantees that, if necessary, the trustworthy organization (OP) can determine or reveal a malicious EV's real identity [18];
5. Perfect forward security: If a long-lasting private key is exposed, the adversary cannot obtain a future session key [19];
6. Perfect backward security: If a long-lasting private key is exposed, an adversary cannot obtain the old session key [19];
7. Joint key control: The session key will be created using a random number that is contributed by both EAG and EV. As a result, no other party has access to or can acquire any session keys;
8. Effective reauthentication: The process where the EAG reauthenticates the EV, causing an overhead. The EAG should, therefore, be able to verify the EV using the information given by a reliable third party (OP) during the initial encounter. Therefore, the EAG does not need to rely on the OP for future access because it can reauthenticate the EV;
9. Revocation method: If a user's registration is ended or the EAG/EV secret key is publicly disclosed, the corresponding information should be revoked. It is critical to grant a revocation mechanism for the system;
10. Attack resistance: Adversaries may launch attacks during the communication between EAG and EV, as it is carried out in an insecure environment. Thus, the proposed scheme must be capable of thwarting attacks such as MITM attacks, replay attacks, impersonation attacks, etc.

2.2. ECQV Implicit Certificates

In comparison to the standard certificate (such as X.509 certificates), the implicit certificate provided by the ECQV method offers the advantages of having a smaller certificate that is computationally quicker and more ideal for IoT devices with limited resources [20].

A conventional certificate needs to have its signature verified, whereas an implicit certificate only needs to have its public key derived, and the latter is quicker than the former. The entity seeking the security material is the only one who can derive the private key; hence it is not even accessible to the certificate authority (CA). Hence, the technique is protected against key escrow attacks. Furthermore, a secure connection is not necessary during the operation because all variables may be delivered via the open channel [21].

(1) ECQV Basic Notations

In Table 1 below, the fundamental notations used in the ECQV scheme are defined.

Table 1. ECQV basic notations.

Notations	Meaning
d_A	EC private key for entity A
Q_A	EC public key for entity A
$k \in_R [1, \ldots, \mathrm{n}-1]$	k integer, a random value between 1 to n − 1
G	Base point in E_p with order n
E_p	Elliptic curve (EC) over a finite field with p being a significant prime number
$H(.)$	One-way hash function
r	Private reconstruction data
P_A	Public reconstruction data
ID_A	Identity of entity A
e	Hash of certificate
$Cert_x$	Certificates of entity x

(2) ECQV Algorithms

ECQV implicit certificates are produced using ECQV technology, which is based on elliptic curve cryptography. The elliptic curve domain settings for this method must be agreed upon by the entities and the CA before it can be used. Figure 1 shows the details of this strategy, which consists of three steps [22]:

1. ECQV certificate request: A user generates an EC pair of keys and sends the public key together with the user's ID to the CA;
2. ECQV certificate generation: The CA validates the ID and creates data for public reconstruction that may be used to obtain the user's public key. Next, ECQV certificate data are incorporated and contain both ID and public reconstruction information. The resulting ECQV certificate and the private key of the CA are then used to compute the user's private reconstruction data. The user then receives the private reconstruction data and the ECQV certificate from the CA;
3. ECQV certificate reception: The user creates a public/private-key pair using the first step's private key, private reconstruction information, and ECQV certificate (acquired from CA). In order to confirm that the obtained certificate was indeed issued by the CA, the user then performs a verification process.

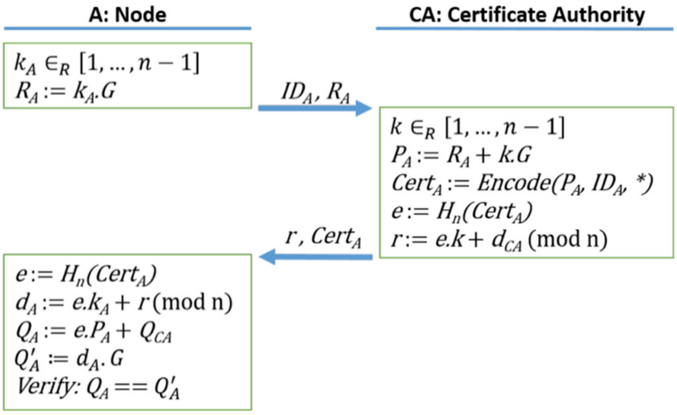

A: Node
CA: Certificate Authority
$k_A \in_R [1, \ldots, n-1]$
$R_A := k_A.G$
ID_A, R_A
$k \in_R [1, \ldots, n-1]$
$P_A := R_A + k.G$
$Cert_A := Encode(P_A, ID_A, *)$
$e := H_n(Cert_A)$
$r := e.k + d_{CA}\ (\mathrm{mod\ n})$
$r, Cert_A$
$e := H_n(Cert_A)$
$d_A := e.k_A + r\ (\mathrm{mod\ n})$
$Q_A := e.P_A + Q_{CA}$
$Q'_A := d_A.G$
$Verify: Q_A == Q'_A$

Figure 1. Generation of implicit certificates for ECQV [22].

IoT devices can create a secure communication channel using ECQV implicit certificates and Elliptic Curve Diffie–Hellman (ECDH) for authenticated key exchange. The process of the ECQV implicit certificate-based authenticated key exchange algorithm is presented in Figure 2.

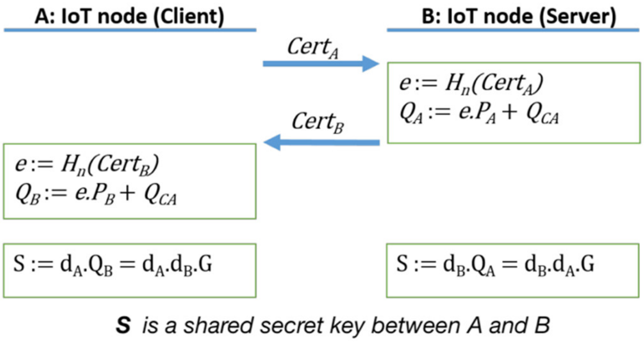


Figure 2. The ECQV implicit certificate-based authenticated key exchange algorithm [22].

3. Literature Review

The most relevant, significant EV charging privacy preservation and security techniques are reviewed in this section. Multiple security systems already in use are discussed, as well as their advantages and disadvantages, along with the efforts this study took to solve the issues. To simplify the authentication process and make it less difficult, this work aims to propose a safe authentication strategy for EV charging that protects the privacy and offers a reliable reauthentication mechanism. Previous works in the security and privacy systems sectors have been reviewed for the intended outcome.

There are two main categories that the authentication protocols fall into: public-key authentication and symmetric-key authentication. Asymmetric cryptography, commonly known as public-key cryptography, is an encryption/decryption technique that utilizes a key pair made up of public/private keys [23]. However, symmetric-key cryptography relies on a single key that may be used for both encryption/decryption [24].

It was recently demonstrated that utilizing only XOR and hash operations in symmetric-key-based authentication protocols [25–27] can ensure anonymity and the un-traceability of a user's behavior. Li et al. (2017) applied symmetric-key cryptography to provide an authentication technique for a dynamic charge system. The technique enables EVs to authenticate anonymously to charging piles (CP) while maintaining their geographical privacy. However, the process of forwarding the credentials (pseudonyms) and the distribution of associated keys to all CPs is inefficient, as it requires a large space to store all these data within CPs and leads to communication overhead. Additionally, an EV's real identity is revealed to the service provider to create the pseudonym identity, which is risky and expensive computationally [25].

For EV charging, a blockchain-based security model was proposed by Huang et al. (2018). It provides mutual authentication through symmetric-key cryptography. However, the lightning network charging mechanism was not efficient, and the system was not able to protect the security of the keys. Moreover, the proposed model lacks privacy-preserving features such as anonymity, un-likability, and traceability. Moreover, the proposed authentication protocol requires a secure channel between EV and CP, which is hard to establish and increases the overall cost of the system [26].

The blockchain-based security framework introduced by Kim et al. (2019) used XOR and hash operations to reduce the computational cost of communication. However, the process of authentication must be repeated for each charging session. Furthermore, due to the requirement of all nodes to solve the mathematical computation, the system suffers from latency as the number of electric vehicles grows. Moreover, it does not provide some required privacy-preserving features including un-linkability and traceability [27].

To guarantee secure communication among the various network components, a combination of asymmetric and symmetric cryptography with simple hashing can be used. ElGhanam et al. (2021) applied this combination to provide a lightweight authentication mechanism that enables legitimate EVs to charge while assuring secure and fair payments. The mechanism resists well-known attacks such as replay attacks, MITM attacks, and impersonation attacks. For privacy preservation, it ensures an EV's real identity, anonymity, and un-linkability through the utilization of pseudonyms for each charging process. However, it only provides partial privacy preservation to the EV, as it fails to ensure traceability. If required, the charging company cannot disclose the true identity of the EV that is acting maliciously or inappropriately. They encouraged other researchers to utilize asymmetric cryptography techniques to reduce the computational costs of their scheme [28].

For dynamic charging systems, Babu et al. (2021) presented a robust elliptic curve cryptography-based authentication mechanism. The proposed scheme can mitigate well-known EV attacks including replay attacks, MITM attacks, impersonation attacks, etc. For privacy preservation, it ensures an EV's anonymity and un-traceability. However, similar to other studies in the literature review, it only provides partial privacy preservation to the EV, as there are no mechanisms for un-linkability, traceability, or reauthentication to cut down on communication costs during the authentication process [29].

To preserve the privacy of vehicles, an EV can be first authenticated to the EVCS using a blind signature [30–32]. A digital signature known as a "blind signature" enables the user to have the signer sign any document without being aware of what it contains [33]. Rabieh and Wei (2017) applied a blind signature along with a hash chain to authenticate EVs for dynamic charging while keeping their identities anonymous and un-linkable to other sessions. This was achieved through the usage of pseudonym tickets (to ensure un-linkability) that are published in the revocation list used for authentication and the need for EVs to authenticate themselves multiple times in all phases of the scheme. Unfortunately, the scheme suffers from latency and requires large storage as the number of EVs increases. Moreover, these pseudonyms are generated by EVs randomly and are unregulated. If an EV generates a repeated pseudonym without its knowledge, it will be rejected at the charging center [30].

A partial blind signature along with a hash chain to authenticate EVs for dynamic charging was proposed by Gunukula et al. (2017) to maintain privacy (anonymity and un-linkability). It guarantees resistance to MITM attacks; however, it does not investigate other attacks that could have a wide impact on the system. Furthermore, the system relies on the bank to verify the validity of charging coins, which in turn causes a delay in the authentication process [31].

Roman and Gondim (2019), proposed an authentication protocol for EV dynamic charging infrastructure based on a blind signature along with a hash chain. It ensures resistance to well-known attacks, anonymity, and un-linkability. On the other hand, the scheme requires a secure channel (as it reveals EVs' real identities) in the ticket-purchasing phase; secure channel establishment is hard and is mainly achieved by physical contact prior to the service request from the charging company. We believe that better solutions exist for handling such requirements [32].

Other techniques used to authenticate EVs and maintain their privacy are public-key, sign-encryption, and group-signature algorithms. They allow signing messages exchanged by EV users, making it impossible for malicious individuals or other network attackers to learn target EVs' real identities. In contrast to the signature-then-encryption method, signcryption is public-key cryptography (PKC) primitive that simultaneously delivers a digital signature and public-key encryption [34].

Xia et al. (2021) applied the concept of group-signature-based authentication to eliminate the disclosure of an EV's identity to entities other than the CA. Fog computing was used to reduce the interaction between EVs and cloud servers. However, the scheme needs to reduce the entities' interaction, as it is considered to be high [35].

Two of the techniques used to improve the protocol's communication cost are partial identity-based signcryption (IBSC) and pairing-based protocol for EV group authentication, as in the protocol proposed by Roman et al. (2019), where a group message is used to protect the anonymity of EVs. The protocol ensures communication confidentiality, location privacy, and resistance to several attacks. However, at the registration stage, a public/private-key pair is generated for the group; due to this process, there is a need for larger storage and the issue of single-group association must be considered (unchanged until EV requests to leave the group). Furthermore, there is no plan in place to guard against the compromise of single-group association [36].

Kumar et al. (2020) introduced a framework for EV charging using signcryption cryptography to ensure security and privacy preservation. It uses pseudo-identity to provide anonymity for EVs and it resists several attacks including replay attacks, MITM attacks, impersonation attacks, etc. However, the scheme does not provide un-linkability, traceability, or a mutual or reauthentication mechanism to reduce the overhead of the authentication process [37].

Public-key infrastructure (PKI) with smart cards or contract certificates was proposed by Vaidya and Mouftah (2020) to authenticate EVs in plug-and-charge systems. The scheme ensures resistance to MITM and impersonation attacks. However, it does not provide assurance against replay and stolen card attacks, and it is feasible to eavesdrop on data exchange by placing fake card reader devices. Moreover, the system lacks the protection of EV privacy. The scheme has low communication, but it lacks security and privacy [38].

Additionally, according to [39], group-signature authentication has some drawbacks that should be taken into account when designing the system, such as how the private key for the group of EVs is distributed, how frequently the public/private-key pairs need to be changed, and how the key management mechanism works.

Many security protection solutions in vehicle-to-grid (V2G) networks currently rely on anonymity, pseudonymity, and encryption technologies. To address the issue of anonymity, existing research and protocols employ traditional processes such as blind signatures and bilinear pairing. The performance–security trade-off is significantly impacted by all of these systems' high processing and communication requirements. Due to the complexity of signature and encryption mechanisms, the time required for authentication between an EV and the power grid would significantly increase (such as blind signatures). Simple anonymity and pseudonym solutions are no longer sufficient in addressing the issue of identification for EV users [40].

Solutions by [26,30,32] require a secure channel to purchase the tickets, which is both hard to establish and costly. PKI is the only way to avoid such a secure channel [38]. Studies [30,32] use certificates to initially register and authenticate and are categorized as token-based authentication schemes, as they use tickets for EV authentication. For security and privacy preservation, various blockchain-based EV charging systems have been developed. Anonymous communication hides an EV's real identity; however, if the same anonymous ID is used multiple times, it threatens the EV's privacy (un-linkability). By linking data to other publicly available datasets such as transactions, data might be utilized to execute privacy-related linkage attacks, and the cloud can carry out attacks using a variety of data-mining techniques and algorithms. However, absolute anonymity is the main privacy feature considered in most of these systems, which is not sufficient in maintaining order in the EV charging infrastructure [41].

As IoT devices tend to be resource-constrained and the applications of vehicular ad hoc networks (VANETs) are latency-critical, Ha et al. (2016) introduced a security scheme based on ECQV implicit authentication. For IoT devices to have mutual authentication, key establishment, and key exchange capabilities, an ECQV implicit certificate is essential. Ha et al. were able to demonstrate through a computational test that ECQV implicit certificates are preferable to traditional certificates for use in IoT devices with limited resources [22]. In their study, Baee et al. (2019) explored how much the authentication overhead in latency-critical apps affects the safety of EV drivers. They also demonstrated that combining the Elliptic Curve Digital Signature Algorithm (ECDSA) and ECQV over the National Institute of Standard and Technology (NIST) P-256 curve and validating certificates can be a viable solution [42].

The earlier authentication techniques did not consider other fundamental privacy-preserving criteria in VANETs, such as un-linkability and traceability. These requirements integrate into each other because the EV's real identity is hidden, different sessions are not linkable, and misbehaving anonymous vehicles are traced by authorities. Additionally, we noticed that the concept of token-based reauthentication for authentic EVs within a short period of time has not been tackled. Our work is the first to suggest using ECQV to authenticate EVs EAGs in the charging system, to our knowledge. In order to ensure that the privacy of EVs is maintained for EV charging systems, we provide a lightweight ECQV-based authentication scheme. It delivers reliable and safe authentication and reauthentication.

4. Proposed System

In this section, we present the proposed authentication protocols that address the solution requirements. Table 2 details the notation used for this scheme's phases: (1) Initialization, (2) Registration, (3) Authentication, and (4) Charging.

Table 2. Notations.

Notations	Meaning
EV	Electric vehicle
EAG	Energy aggregator
OP	Electricity operator
E_p	Elliptic curve (EC) over a finite field, with p being a significant prime integer
G	E_p's base point with order n
id_{EV}, id_{EAG}	EV/EAG's true identity
$k_x,\ R_x$	Pair of EC keys for entity x
S_x	Data used to construct entity x‘s private key
$Cert_x$	Entity x's certificate
$Sig_x\ (y)$	Message y is signed by entity x using x's private key
$PK_x\ (y)$	Using entity x's public key, entity x encrypts message y
A_x	Entity x's authenticator
AH_x	Hash of entity x's authenticator
T_x	Time stamp produced by x
TL	Time-life
e	Certificate hash
$PK_x,\ PR_x$	Entity x's Public/Private-key pair
RK, RK'	EV and OP/EAG and OP registration key
Aid_i	EV′s i^{th} anonymous identity established by OP
Aid_{No}	Aid_i counter that is incremented by EV
N_x	Nonce by x
AT_{EV}^{EAG}	EV's authorization token, generated by EAG
K_{EV-EAG}	EV and EAG shared symmetric master key
IK_{EV-EAG}	EV and EAG shared symmetric initial key
TK_{EV-EAG}	EV and EAG shared symmetric temporary key
SK_{EV-EAG}	EV and EAG shared symmetric session key
$H(.)$	One-way hash function
(y, x)	Concatenation operation

4.1. System Architecture

The entities electricity operator (OP), energy aggregator (EAG), and electric vehicles (EVs) compose our scheme. Figure 3 shows the architecture, and the list of entities in our design are listed below:

1. Operator (OP): Any EV or EAG seeking to use the charging system must first register their identification information with the OP, where OP acts as the initializer for the proposed protocol. Authorized EVs can use the EAG's services and develop trust with one another because the OP acts as a certificate generator (trusted third party). The OP can also identify malicious or misbehaving nodes by revealing their identities;
2. Energy Aggregator (EAG): A data aggregator is a smart device or collection of smart devices that serves as a data aggregator of available EV power information while the EVs are charging and supplying power to the EVs via a number of EVCSs. To

coordinate the charging, the EAG has an authentication mechanism to identify authorized EVs;

3. *Electric Vehicle (EV):* It is a smart device that communicates charging requests to EAGs and mutually validates an EAG's eligibility to use its service (charging).

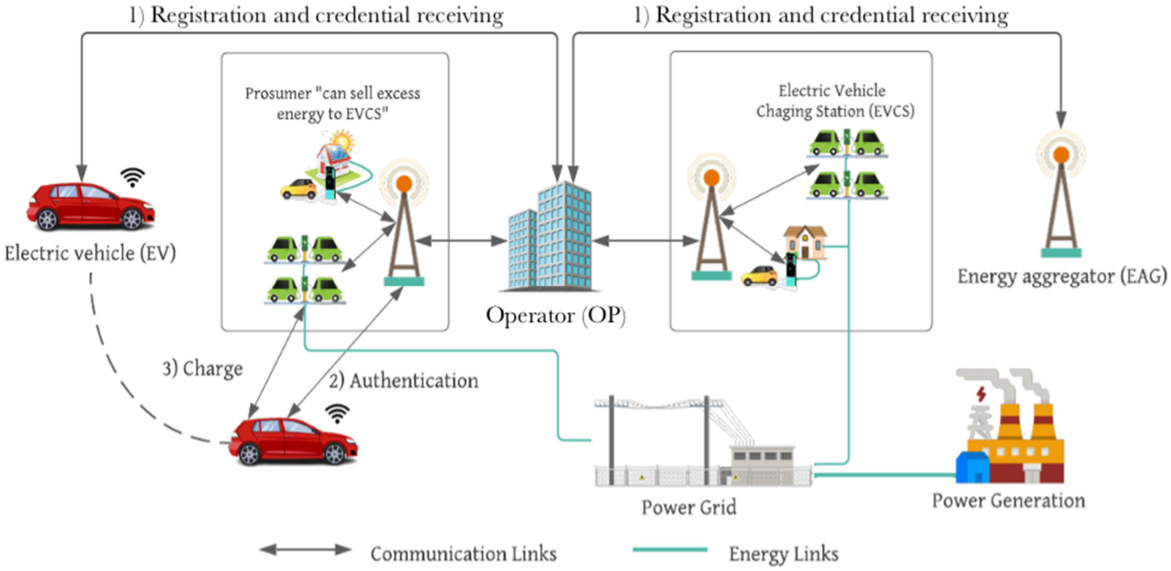

1) Registration and credential receiving
1) Registration and credential receiving
Prosumer "can sell excess energy to EVCS"
Electric Vehicle Chaging Station (EVCS)
Electric vehicle (EV)
Operator (OP)
Energy aggregator (EAG)
3) Charge
2) Authentication
Power Grid
Power Generation
Communication Links
Energy Links

Figure 3. Proposed system architecture.

From the architecture in Figure 3, the first step in establishing the communication between an EV and an EAG is the registration with an OP. After receiving credentials, these entities become part of the system and can communicate and authenticate each other if necessary. After successful verification, the EV can access a charging service. The EAG authenticates the EV using the A_{EV} signed by the OP and Aid_i by the EV, without the direct involvement of the OP. The other way around, the EV authenticates the EAG using A_{EAG} signed by the OP to thwart impersonation attacks. The proposed solution makes use of symmetric, ECQV, and PKC methodologies to ensure secure communication and to shorten the computation time in order to establish mutual authentication between the EV and EAG. For effective reauthentication, the proposed solution permits the reuse of AT_{EV}^{EAG} (similar to [43] and our previous work [44]) and utilizes the speed constraint in [11] to countermeasure location-related attacks. AT_{EV}^{EAG} is issued by the EAG once it has authenticated the EV. The utilization of AT_{EV}^{EAG} reduces the time required to verify A_{EV} in upcoming charging requests. Since there are not enough public EVCSs, which is the main problem with the EV charging infrastructure, EV owners may help other EVs in need by lending them their personal charging stations. In return, personal EVCS owners can earn some incentives through sharing with other EVs or by selling excess power to OPs.

4.2. Threat Model

The EAG is responsible for energy node matching and providing location-based services to electric vehicle owners during the energy trading process. We assume that the location-related communication is not secure enough (internet, Wi-Fi, Bluetooth, dedicated short-range communications (DSRC), etc.), and adversaries can use these services to determine the precise location of the target EV owner. Since EV owners' location data includes vital information such as their house, workplace, hospitals, and so on, once learned by the adversary, the privacy of EV owners will be compromised, and their personal safety may be jeopardized.

The two types of attackers that are interested in obtaining the location data and credentials of EV owners are outsider and insider attackers. The transaction data collected by the system can be used by an outsider attacker to obtain information about the location of EV owners; a malicious internal node in our proposed scheme acts as an insider attacker and can gather user data. We assume that the EAG is a potential insider threat who is capable of learning the credentials or precise location of EV owners throughout the trading process.

4.3. Initialization Phase

The following steps describe how the OP initializes the system to set up the network:

Step 1: A base point G of order n is chosen by OP on the elliptic curve E_p, where n is a significant prime number. Select the curve coefficients a and b, field size q, and cofactor h, where hn is the number of points on the elliptic curve (these are the elliptic curve domain parameters);

Step 2: Select an approved hash function $H(.)$. The OP and certificate requester (EV or EAG) specify the generator of random numbers to be used throughout the certificate request/creation procedures to generate the private keys;

Step 3: OP obtains an EC-key pair (PR_{OP}, PK_{OP}), which is associated with the elliptic curve domain parameters (established in the first step).

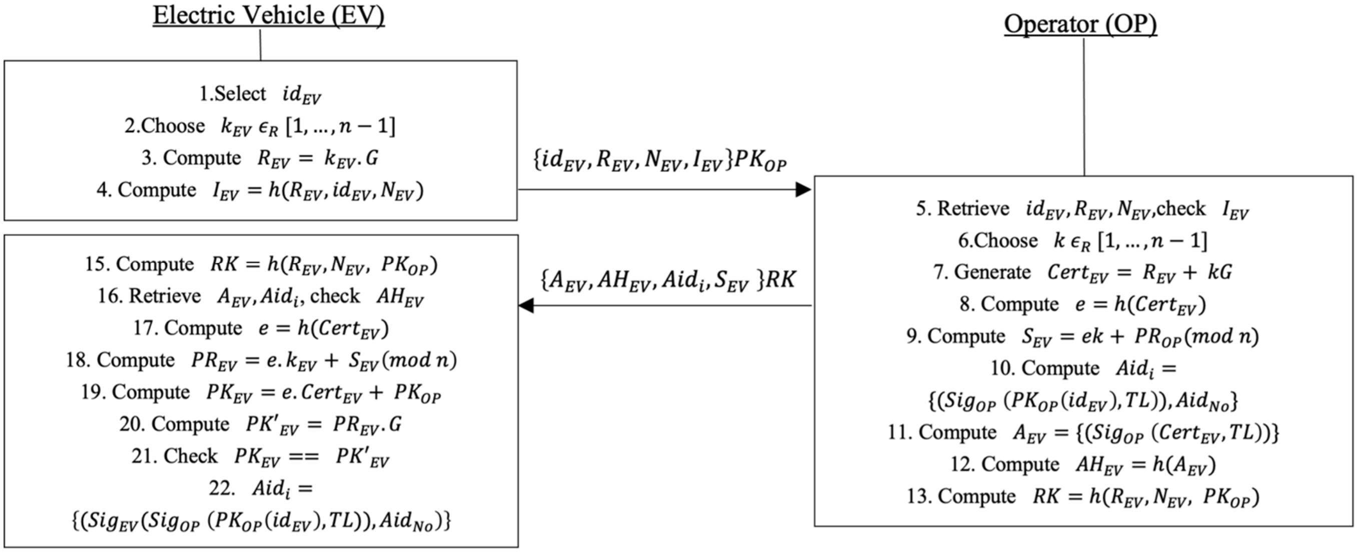
Step 4: Both EV and EAG obtain, in an authentic manner, the EC domain parameters, $H(.)$, and PK_{OP} (OP's public key).

4.4. Registration Phase

Before EVs and EAGs can join the charging system, they must produce their identities and pair of public/private keys. After which, they receive the corresponding certificate, authenticator, information about constructing the private key, and anonymous identity from the OP (for EV only).

4.4.1. EV Registration

The registration process for EVs is presented in Figure 4 and is detailed as follows:



Electric Vehicle (EV)
Operator (OP)
1.Select id_{EV}
2.Choose $k_{EV}\,\epsilon_R\,[1,\dots,n-1]$
3. Compute $R_{EV} = k_{EV}.G$
4. Compute $I_{EV} = h(R_{EV}, id_{EV}, N_{EV})$
$\{id_{EV}, R_{EV}, N_{EV}, I_{EV}\}PK_{OP}$
5. Retrieve id_{EV}, R_{EV}, N_{EV},check I_{EV}
6.Choose $k\,\epsilon_R\,[1,\dots,n-1]$
7. Generate $Cert_{EV} = R_{EV} + kG$
8. Compute $e = h(Cert_{EV})$
9. Compute $S_{EV} = ek + PR_{OP}(mod\,n)$
10. Compute $Aid_i =$
$\{(Sig_{OP}\,(PK_{OP}(id_{EV}), TL)), Aid_{No}\}$
11. Compute $A_{EV} = \{(Sig_{OP}\,(Cert_{EV}, TL))\}$
12. Compute $AH_{EV} = h(A_{EV})$
13. Compute $RK = h(R_{EV}, N_{EV},\ PK_{OP})$
$\{A_{EV}, AH_{EV}, Aid_i, S_{EV}\,\}RK$
15. Compute $RK = h(R_{EV}, N_{EV},\ PK_{OP})$
16. Retrieve A_{EV}, Aid_i, check AH_{EV}
17. Compute $e = h(Cert_{EV})$
18. Compute $PR_{EV} = e.k_{EV} + S_{EV}(mod\,n)$
19. Compute $PK_{EV} = e.Cert_{EV} + PK_{OP}$
20. Compute $PK'_{EV} = PR_{EV}.G$
21. Check $PK_{EV} == PK'_{EV}$
22. $Aid_i =$
$\{(Sig_{EV}(Sig_{OP}\,(PK_{OP}(id_{EV}), TL)), Aid_{No})\}$

Figure 4. The proposed EV registration.

Step 1: EV selects its identity id_{EV}; generate EC-key pair (k_{EV}, R_{EV}), where k_{EV} ϵ_R $[1, \dots, n-1]$ and $R_{EV} = k_{EV}.G$. Compute $I_{EV} = h(R_{EV}, id_{EV}, N_{EV})$ to ensure integrity. Then, send it to OP $\{id_{EV}, R_{EV}, N_{EV}, I_{EV}\}$ encrypted with PK_{OP} (OP's public key);

Step 2: OP retrieves the content of the message using its private key PR_{OP} and verifies I_{EV}. Then, choose k ϵ_R $[1, \dots, n-1]$ and generate EV's implicit certificate $Cert_{EV} = R_{EV} + kG$; compute $e = h(Cert_{EV})$, $S_{EV} = ek + PR_{OP}(mod\ n)$, the private key construction data of EV. OP uses Formula (1) to create EV's pseudo-identity and signs it using OP's private key, where the real identity of EV is encrypted with PK_{OP} to assure its anonymity and Aid_i is agreed to be incremented sequentially (Aid_{No}) by EV itself every time it requests a service. Compute EV's authenticator using Formula (2), which contains the issued $Cert_{EV}$ with its time-life (TL) signed using the private key of OP. Compute $AH_{EV} = H(A_{EV})$ to ensure integrity. Compute registration key $RK = h(R_{EV}, N_{EV},\ PK_{OP})$ that is shared between OP and EV only. Then, send to EV $\{A_{EV},\ AH_{EV},\ Aid_i, S_{EV}\}$ encrypted with RK. Lastly, OP destroys R_{EV}, k, S_{EV} to prevent the possession of EV's private key by an adversary;

$$Aid_i = \{(Sig_{OP}\ (PK_{OP}(id_{EV}), TL)),\ Aid_{No}\} \tag{1}$$

$$A_{EV} = \{(Sig_{OP}\ (Cert_{EV}, TL,\ Aid_i)) \tag{2}$$

Step 3: EV calculates the shared registration key $RK = h(R_{EV}, N_{EV},\ PK_{OP})$ to retrieve and verify $A_{EV},\ Aid_i$ through OP's public key and check AH_{EV}. Compute $e = h(Cert_{EV})$ to generate its private/public-key pair PR_{EV}/PK_{EV} using Formulas (3) and (4).

$$PR_{EV} = e.k_{EV} + S_{EV}(mod\ n) \tag{3}$$

$$PK_{EV} = e.Cert_{EV} + PK_{OP} \tag{4}$$

To ensure the validity of PK_{EV}, it computes $PK'_{EV} = PR_{EV}.G$; then, check if $PK_{EV} == PK'_{EV}$ as follows:

$$\begin{aligned} PR_{EV} &= e.k_{EV} + S_{EV}(mod\ n) \\ &= e.k_{EV} + (e.k + PR_{OP}(mod\ n)) \\ &= e.(k_{EV} + k) + PR_{OP}(mod\ n)) \end{aligned} \tag{5}$$

$$\begin{aligned} Cert_{EV} &= R_{EV} + kG \\ &= k_{EV}.G + k.G \\ &= (k_{EV} + k).G \end{aligned} \tag{6}$$

$$\begin{aligned} PK_{EV} &= e.Cert_{EV} + PK_{OP} \\ &= e.(k_{EV} + k).G + PR_{OP}.G \\ &= e.((k_{EV} + k) + PR_{OP}).G \\ &= PR_{EV}\ .G \end{aligned} \tag{7}$$

After the validation of $PK_{EV} == PK'_{EV}$, EV adds its signature to Aid_i using its own private key. Lastly, EV stores $\{A_{EV}\ ,\ Aid_i\}$ encrypted with PK_{EV} in memory (on-board unit—OBU). Destroy R_{EV}, k_{EV}, S_{EV} to prevent the possession of an EV's private key by an adversary.

4.4.2. EAG Registration

The registration process for the EAG is presented in Figure 5 and is detailed as follows:

Step 1: EAG selects its identity id_{EAG}; generate EC-key pair ($k_{EAG},\ R_{EAG}$), where $k_{EAG}\ \epsilon_R\ [1, \ldots, n-1]$ and $R_{EAG} = k_{EAG}.G$. Compute $I_{EAG} = h(R_{EAG},\ id_{EAG},\ N_{EAG})$ to ensure integrity. Then, send it to OP $\{id_{EAG},\ R_{EAG},\ N_{EAG},\ I_{EAG}\}$ encrypted with PK_{OP}.

Step 2: OP retrieves the content of the message using its private key PR_{OP} and verifies I_{EAG}; choose $k\ \epsilon_R\ [1, \ldots, n-1]$ and generate EAG's implicit certificate $Cert_{EAG} = R_{EAG} + kG$; compute $e = h(Cert_{EAG})$, $S_{EAG} = ek + PR_{OP}(mod\ n)$, the private key construction data of EAG. Compute the authenticator by Formula (8); it contains the issued $Cert_{EAG}, id_{EAG}$, its time-life (TL) signed using the private key of OP. Then, compute $AH_{EAG} = H(A_{EAG})$ to ensure integrity. Compute registration key $RK' = h(R_{EAG}, N_{EAG},\ PK_{OP})$ that is shared between OP and EAG only. Then, send it to EAG $\{A_{EAG},\ AH_{EAG}, S_{EAG}\}$ encrypted with RK'. Lastly, OP destroys R_{EAG}, k, S_{EAG} to prevent the possession of EAG's private key by adversaries.

$$A_{EAG} = \{(Sig_{OP}(Cert_{EAG},\ TL,\ id_{EAG})\} \tag{8}$$

Step 3: EAG computes the registration key $RK' = h(R_{EAG}, N_{EAG},\ PK_{OP})$ to retrieve and verify the A_{EAG} through OP's public key and checks AH_{EAG}. Compute $e = h(Cert_{EAG})$ to generate its private/public-key pair PR_{EAG}/PK_{EAG} using Formulas (9) and (10).

$$PR_{EAG} = e.k_{EAG} + S_{EAG}(mod\ n) \tag{9}$$

$$PK_{EAG} = e.Cert_{EAG} + PK_{OP} \tag{10}$$

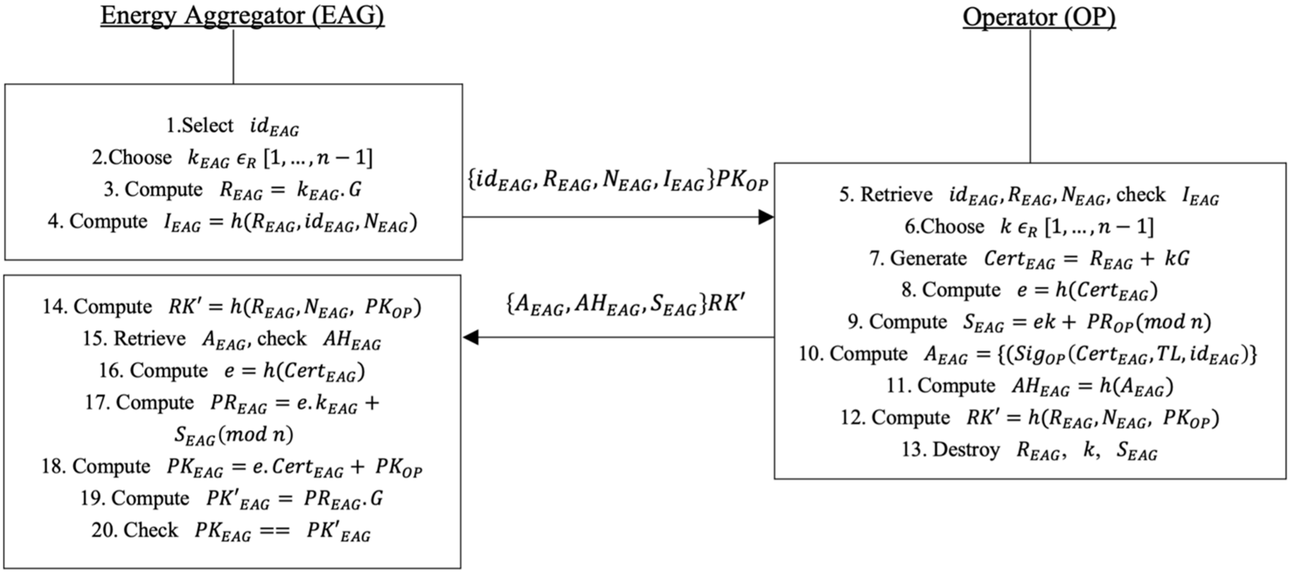

Energy Aggregator (EAG)
Operator (OP)
1.Select id_{EAG}
2.Choose $k_{EAG}\ \epsilon_R\ [1, \ldots, n-1]$
3. Compute $R_{EAG} = k_{EAG}.G$
4. Compute $I_{EAG} = h(R_{EAG}, id_{EAG}, N_{EAG})$
$\{id_{EAG}, R_{EAG}, N_{EAG}, I_{EAG}\}PK_{OP}$
5. Retrieve $id_{EAG}, R_{EAG}, N_{EAG}$, check I_{EAG}
6.Choose $k\ \epsilon_R\ [1, \ldots, n-1]$
7. Generate $Cert_{EAG} = R_{EAG} + kG$
8. Compute $e = h(Cert_{EAG})$
9. Compute $S_{EAG} = ek + PR_{OP}(mod\ n)$
10. Compute $A_{EAG} = \{(Sig_{OP}(Cert_{EAG}, TL, id_{EAG})\}$
11. Compute $AH_{EAG} = h(A_{EAG})$
12. Compute $RK' = h(R_{EAG}, N_{EAG},\ PK_{OP})$
13. Destroy $R_{EAG},\ k,\ S_{EAG}$
$\{A_{EAG}, AH_{EAG}, S_{EAG}\}RK'$
14. Compute $RK' = h(R_{EAG}, N_{EAG},\ PK_{OP})$
15. Retrieve A_{EAG}, check AH_{EAG}
16. Compute $e = h(Cert_{EAG})$
17. Compute $PR_{EAG} = e.k_{EAG} + S_{EAG}(mod\ n)$
18. Compute $PK_{EAG} = e.Cert_{EAG} + PK_{OP}$
19. Compute $PK'_{EAG} = PR_{EAG}.G$
20. Check $PK_{EAG} == PK'_{EAG}$

Figure 5. The proposed EAG registration phase.

To ensure the validity of PK_{EAG}, it computes $PK'_{EAG} = PR_{EAG}.G$; then, check if $PK_{EAG} == PK'_{EAG}$ as follows:

$$\begin{aligned} PR_{EAG} &= e.k_{EAG} + S_{EAG}(mod\ n) \\ &= e.k_{EAG} + (e.k + PR_{OP}(mod\ n)) \\ &= e.(k_{EAG} + k) + PR_{OP}(mod\ n)) \end{aligned} \tag{11}$$

$$\begin{aligned} Cert_{EAG} &= R_{EAG} + kG \\ &= k_{EAG}.G + k.G \\ &= (k_{EAG} + k).G \end{aligned} \tag{12}$$

$$\begin{aligned} PK_{EAG} &= e.Cert_{EAG} + PK_{OP} \\ &= e.(k_{EAG} + k).G + PR_{OP}.G \\ &= e.((k_{EAG} + k) + PR_{OP}).G \\ &= PR_{EAG}\ .G \end{aligned} \tag{13}$$

After the validation of $PK_{EAG} == PK'_{EAG}$, EAG stores $\{A_{EAG}\}$ encrypted with PK_{EAG} in memory and destroys R_{EAG}, k_{EAG}, S_{EAG} to prevent the possession of EAG's private key by an adversary.

4.5. Authentication Phase

When an EV wishes to access the charging system, the EV and EAG must authenticate one another and create a session key. The authentication phase is separated into two groups: mutual authentication, in which EV and EAG have not yet developed a relationship of trust. Thus, they rely on the information provided by the OP (third party). The second is lightweight reauthentication, where the EV and EAG authenticate each other on their own without a third trusted party (OP).

4.5.1. Mutual Authentication Protocol

This process is conducted initially, where the EAG relays on the OP's information to authenticate the EV, unless A_{EV} is terminated. Figure 6 illustrates the procedure and provides the following details:

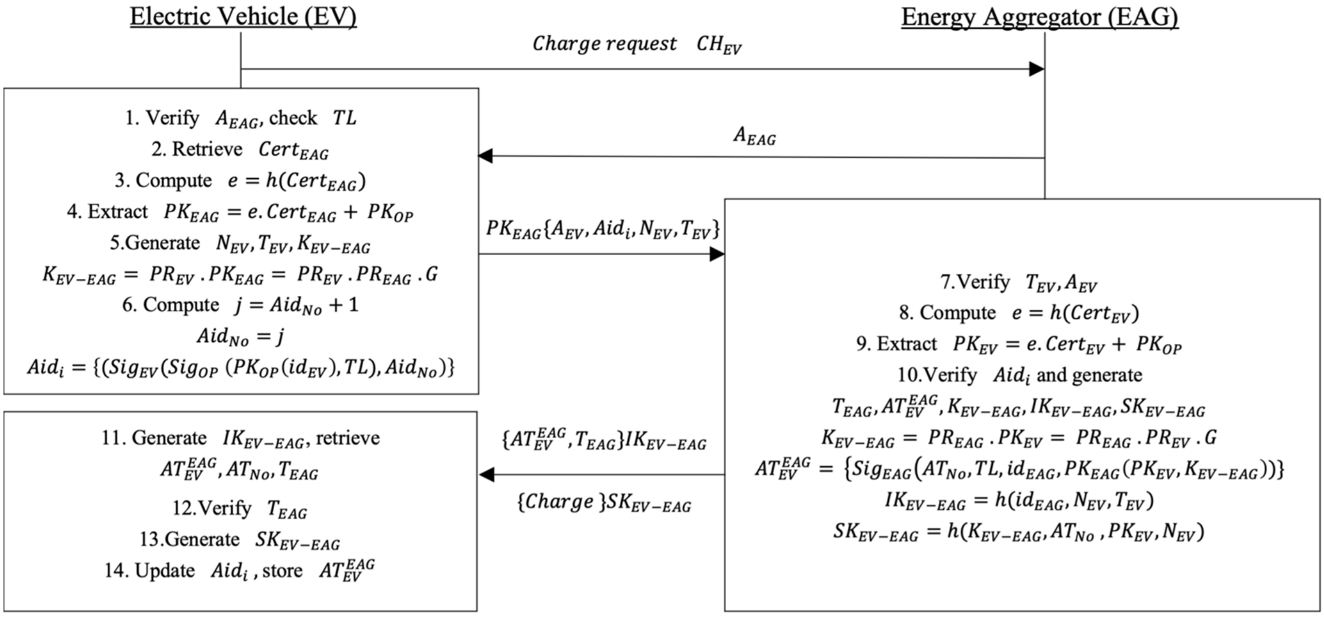

Electric Vehicle (EV)
Energy Aggregator (EAG)
Charge request CH_{EV}
1. Verify A_{EAG}, check TL
2. Retrieve $Cert_{EAG}$
3. Compute $e = h(Cert_{EAG})$
4. Extract $PK_{EAG} = e.Cert_{EAG} + PK_{OP}$
5.Generate $N_{EV}, T_{EV}, K_{EV-EAG}$
$K_{EV-EAG} = PR_{EV} . PK_{EAG} = PR_{EV} . PR_{EAG} . G$
6. Compute $j = Aid_{No} + 1$
$Aid_{No} = j$
$Aid_i = \{(Sig_{EV}(Sig_{OP}(PK_{OP}(id_{EV}), TL), Aid_{No})\}$
A_{EAG}
$PK_{EAG}\{A_{EV}, Aid_i, N_{EV}, T_{EV}\}$
7.Verify T_{EV}, A_{EV}
8. Compute $e = h(Cert_{EV})$
9. Extract $PK_{EV} = e.Cert_{EV} + PK_{OP}$
10.Verify Aid_i and generate
$T_{EAG}, AT_{EV}^{EAG}, K_{EV-EAG}, IK_{EV-EAG}, SK_{EV-EAG}$
$K_{EV-EAG} = PR_{EAG} . PK_{EV} = PR_{EAG} . PR_{EV} . G$
$AT_{EV}^{EAG} = \{Sig_{EAG}(AT_{No}, TL, id_{EAG}, PK_{EAG}(PK_{EV}, K_{EV-EAG}))\}$
$IK_{EV-EAG} = h(id_{EAG}, N_{EV}, T_{EV})$
$SK_{EV-EAG} = h(K_{EV-EAG}, AT_{No}, PK_{EV}, N_{EV})$
$\{AT_{EV}^{EAG}, T_{EAG}\}IK_{EV-EAG}$
$\{Charge\}SK_{EV-EAG}$
11. Generate IK_{EV-EAG}, retrieve
$AT_{EV}^{EAG}, AT_{No}, T_{EAG}$
12.Verify T_{EAG}
13.Generate SK_{EV-EAG}
14. Update Aid_i , store AT_{EV}^{EAG}

Figure 6. The proposed authentication phase.

Step 1: EV generates the charging request $CH_{EV} = \{amount,\ price, distance, TL\}$, where "*amount*" states the amount of power needed, "*price*" specifies how much the EV is willing to pay for the service (to keep the location of the EV private), "*distance*" should specify how far it is from the local EAG, and "TL" states the time-life of the request. Then, EV sends CH_{EV} to their local EAG;

Step 2: The EAG sends its $A_{EAG},\ AH_{EAG}$ as a response to the EV charging request;

Step 3: EV verifies A_{EAG} through OP's signature and checks if it is valid by the TL; retrieve $Cert_{EAG}$ and compute $e = h(Cert_{EAG})$ to extract the EAG's public key $PK_{EAG} = e.Cert_{EAG} + PK_{OP}$. Generate a random number N_{EV}, time stamp T_{EV}, and the master shared key K_{EV-EAG} using Formula (14). Increase the Aid_i counter one at a time (by adding 1 to the previous EV's pseudo-identification) to generate a new anonymous identity for the current session. This prevents linking between multiple sessions of an EV. Then, send to EAG $\{A_{EV}, Aid_i,\ N_{EV},\ T_{EV}\}$ encrypted with PK_{EAG}.

$$K_{EV-EAG} = PR_{EV}\ .\ PK_{EAG} = PR_{EV}\ .\ PR_{EAG}\ .G \tag{14}$$

Step 4: To decrypt the message, EAG employs its own private key and uses OP's signature to confirm that A_{EV} is authentic, and checks T_{EV} to make sure the message is not being replayed. Compute $e = H(Cert_{EV})$ to extract EV's public key $PK_{EV} = e.Cert_{EV} + PK_{OP}$. Verify Aid_i by OP's signature and EV's signature (PK_{EV}) that is included in it. Then, generate N_{EAG}, T_{EAG}, the master shared key using

Formula (15), and the authorization token by Formula (16), where Aid_i, K_{EV-EAG} are encrypted with PK_{EAG}. Moreover, generate the initial and session key using Formulas (17) and (18), receptively. Then, send to EV $\{AT_{EV}^{EAG},\ T_{EAG}\}$ encrypted by IK_{EV-EAG}, the EAG schedule charging service for EVs that is protected by SK_{EV-EAG}.

$$K_{EV-EAG} = PR_{EAG}\ .\ PK_{EV} = PR_{EAG}\ .\ PR_{EV}\ .G \tag{15}$$

$$AT_{EV}^{EAG} = \left\{Sig_{EAG}\left(AT_{No},\ TL,\ id_{EAG},\ PK_{EAG}(PK_{EV},\ K_{EV-EAG})\right)\right\} \tag{16}$$

$$IK_{EV-EAG} = h(id_{EAG},\ N_{EV},\ T_{EV}) \tag{17}$$

$$SK_{EV-EAG} = h(K_{EV-EAG},\ AT_{No},\ \ PK_{EV},\ N_{EV}) \tag{18}$$

Step 5: EV generates IK_{EV-EAG} by Formula (17) to retrieve AT_{EV}^{EAG}, AT_{No} and checks the validity of T_{EAG}. Next, create the session key to be used during the charging session using Formula (18). EV stores the AT_{EV}^{EAG} issued by EAG, and updates Aid_i. By the end of this process, EV and EAG shall have both established trust between them, without having to depend on OP in the future for session authentication.

4.5.2. Lightweight Mutual Reauthentication Protocol

As noted before, at this point, the EV and EAG should have developed a relationship of trust. Now, they can directly and mutually authenticate one another in upcoming charging services. Since charging is a rapidly needed service and a matching process is vital in providing the service, reauthentication can guarantee faster matching for the EV to obtain the service faster. When the user has a valid AT_{EV}^{EAG} and is scheduled to the same aggregator within a 48-h period, this phase (reauthentication phase) can be utilized. The reauthentication time window is chosen on the basis of EVs' frequent need for recharging and due to EV memory constraints. Furthermore, if the user is already trusted, it is impractical and expensive to generate all the variables for a new session key. The process of efficient reauthentication is presented in Figure 7, detailed as follows:

Step 1: EV creates N_{EV}, N'_{EV}, and applies Formula (18) to determine the previous session key to be used in Aid_i, N'_{EV}, T_{EV} encryption. Increment the Aid_i counter sequentially (add 1 to the EV's previous pseudo-identity) to have a new anonymous identity for this session to maintain un-linkability. Then, EV sends AT_{EV}^{EAG}, N_{EV}, $\{N'_{EV}\}SK_{EV-EAG}$ to EAG.

Step 2: EAG validates the authenticity of AT_{EV}^{EAG} via the signature Sig_{EAG} using PK_{EAG} and TL. Decrypt the AT_{EV}^{EAG} using PR_{EAG} to retrieve PK_{EV}, K_{EV-EAG}. EAG needs to compute SK_{EV-EAG} in order to obtain Aid_i, N'_{EV}, T_{EV}, and confirm that the AT_{EV}^{EAG} was transmitted by the authorized EV. Then, use N'_{EV}, T_{EV}, Aid_{No}, AT_{No} to generate the temporary key TK_{EV-EAG} using Formula (19). Generate N_{EAG}, T_{EAG}, a fresh session key SK'_{EV-EAG} as in the Formula (20). EAG then sends $\{N_{EAG},\ T_{EAG}\}$ encrypted by TK_{EV-EAG} to EV. The EAG manages the EV charging service that is secured by SK'_{EV-EAG}.

$$TK_{EV-EAG} = h\left(AT_{No},\ Aid_{No},\ N'_{EV},\ T_{EV}\right) \tag{19}$$

$$SK'_{EV-EAG} = h(TK_{EV-EAG},\ K_{EV-EAG},\ N_{EAG},\ PK_{EV}) \tag{20}$$

Step 3: EV generates the TK_{EV-EAG} to retrieve N_{EAG} and verifies T_{EAG}. Then, use Formula (20) to create SK'_{EV-EAG} for the charging session; Aid_i is updated.

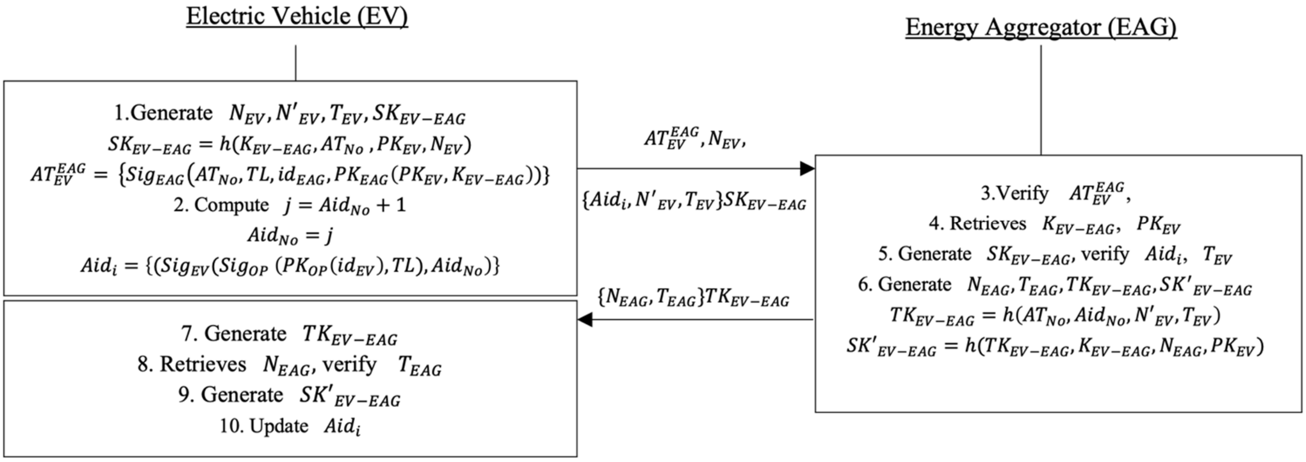

Electric Vehicle (EV)
Energy Aggregator (EAG)
1.Generate $N_{EV}, N'_{EV}, T_{EV}, SK_{EV-EAG}$
$SK_{EV-EAG} = h(K_{EV-EAG}, AT_{No}, PK_{EV}, N_{EV})$
$AT_{EV}^{EAG} = \{Sig_{EAG}(AT_{No}, TL, id_{EAG}, PK_{EAG}(PK_{EV}, K_{EV-EAG}))\}$
2. Compute $j = Aid_{No} + 1$
$Aid_{No} = j$
$Aid_i = \{(Sig_{EV}(Sig_{OP}(PK_{OP}(id_{EV}), TL), Aid_{No})\}$
$AT_{EV}^{EAG}, N_{EV},$
$\{Aid_i, N'_{EV}, T_{EV}\}SK_{EV-EAG}$
3.Verify $AT_{EV}^{EAG},$
4. Retrieves K_{EV-EAG}, PK_{EV}
5. Generate SK_{EV-EAG}, verify Aid_i, T_{EV}
6. Generate $N_{EAG}, T_{EAG}, TK_{EV-EAG}, SK'_{EV-EAG}$
$TK_{EV-EAG} = h(AT_{No}, Aid_{No}, N'_{EV}, T_{EV})$
$SK'_{EV-EAG} = h(TK_{EV-EAG}, K_{EV-EAG}, N_{EAG}, PK_{EV})$
$\{N_{EAG}, T_{EAG}\}TK_{EV-EAG}$
7. Generate TK_{EV-EAG}
8. Retrieves N_{EAG}, verify T_{EAG}
9. Generate SK'_{EV-EAG}
10. Update Aid_i

Figure 7. The proposed reauthentication phase.

4.6. Revocation Protocol

The scheme offers a revocation mechanism to protect the entities from malicious impersonation and MITM attacks and announce that the parameters are no longer reliable even before the validity period has expired. Aid_i and AT_{EV}^{EAG} tokens are revoked in case the EV suspects they were stolen by an adversary, with recency proof (RP) confirming the legitimacy of the OP (e.g., month-old time-stamp proof). The process of Aid_i revocation is detailed as follows:

Step 1: EV creates the Aid_i revocation request $Rev_{EV-id} = \{PK_{OP}(Sig_{EV}, Aid_i,\ RevAid,\ T_{EV})\}$; forward it to OP after being encrypted with OP's public key PK_{OP}.

Step 2: OP decrypts the revocation request by its PR_{OP} and verifies Sig_{EV} using PK_{EV} and Sig_{OP}, which is within Aid_i, and to avoid replay attack, OP checks T_{EV} to verify whether it is valid or not. Finally, OP updates the Aid_i status as revoked. A fake revocation request cannot be produced by the adversary since EV's signature is necessary.

The process of AT_{EV}^{EAG} revocation is detailed below:

Step 1: EV uses Formula (21) to create the AT_{EV}^{EAG} revocation request, which is subsequently sent to EAG after being partially encrypted using Formula (22).

$$Rev_{EV-AT} = \left\{AT_{EV}^{EAG}, N_{EV},\ VK_{EV-EAG}(Sig_{EV}, T_{EV},\ RevAT,\ AT_{No})\right\} \tag{21}$$

$$VK_{EV-EAG} = h(K_{EV-EAG},\ AT_{No},\ N_{EV}) \tag{22}$$

Step 2: EAG validates the AT_{EV}^{EAG} through the signature using PK_{EAG} and decrypts internal part $PK_{EAG}(PK_{EV},\ K_{EV-EAG})$ using its PR_{EAG}. Then, EAG uses K_{EV-EAG}, AT_{No}, N_{EV}, to generate the revocation key VK_{EV-EAG} using Formula (22) and retrieving the other part of the message. Verify the request belongs to the same AT_{EV}^{EAG} by AT_{No} and Sig_{EV} using the retrieved PK_{EV}, then check whether T_{EV} is valid or not. Finally, EAG updates AT_{EV}^{EAG} status as revoked. The use of revoked AT_{EV}^{EAG} leads to the rejection of EV's charging service request. Furthermore, since the master key K_{EV-EAG} is used to construct the revocation key VK_{EV-EAG}, an adversary cannot produce a fake revocation request.

5. Security Analysis

We discuss the proposed protocol's security analysis as well as formal/informal analysis in this section.

5.1. Formal Security Analysis BAN Logic

In authentication protocols, trust connections are evaluated using authentication logic. BAN logic [45] is a frequently used technique for verifying authentication protocols. The proposed protocols' authentication goals will be examined and verified using this logic (Table 3). A protocol analysis utilizing BAN logic can be broken down into four steps for each given protocol:

- Clearly state the goals to achieve;
- Form assumptions about the initial situation;
- Affirm the protocol in its idealized state;
- Utilize the logic to obtain associated party beliefs.

Table 3. BAN logic notations.

Notations	Description
$P \mid\equiv X$	Principal *P believes* statement X is true.
#(X)	Statement X is fresh.
$P \mid\Rightarrow$ X	P has jurisdiction over statement X.
$P \lhd$ X	P sees X, indicating that P has received statement X and could read it.
$P \mid\sim$ X	P once said the statement X.
(X, Y)	The formula (X, Y) includes the terms X or Y.
$\langle X\rangle_Y$	X combined with Y.
$\{X, Y\}_K$	The key K is used to encrypt either X or Y.
$(X, Y)_K$	The key K is used to hash X or Y.
$P \overset{K}{\leftrightarrow} Q$	K is a secret parameter that P and Q share (or will share).
$\xrightarrow{PK_X} X$	Entity X's public key.

The actions and messages of the participating individuals should first be converted into formulas in order to employ the BAN logic. The essential rules for BAN logic are as follows:

Rule1 (Message-meaning rule):

$$R1 = \frac{P \Big|\equiv P \overset{K}{\leftrightarrow} Q,\ \text{P} \lhd \langle X\rangle_Y}{P \mid\equiv Q \mid \sim X} \tag{23}$$

Rule2 (Nonce-verification rule):

$$R2 = \frac{P \mid\equiv \#(X), P \mid \equiv Q \mid\sim X}{P \mid\equiv Q \mid \equiv \text{X}} \tag{24}$$

Rule3 (Jurisdiction rule):

$$R3 = \frac{P \mid\equiv Q \mid\Rightarrow X, P \mid \equiv Q \mid\equiv X}{P \mid\equiv X} \tag{25}$$

Rule4 (Freshness-conjuncatenation rule):

$$R4 = \frac{P \mid\equiv \#\,(X)}{P \mid\equiv \#\,(X, Y)} \tag{26}$$

Rule5 (Belief rule):

$$R5 = \frac{P \mid\equiv (X), P \mid \equiv (Y)}{P \mid\equiv (X, Y)} \tag{27}$$

Rule6 (Session keys rule):

$$R6 = \frac{P \mid\equiv \#(X), P \mid\equiv Q \mid \equiv X}{P \Big|\equiv P \ \overset{K}{\leftrightarrow} \ Q} \tag{28}$$

5.1.1. Analyzing Authentication Protocol

The steps listed below are taken to show that the proposed authentication protocol is accurate.

Step 1: Goals. The analysis' key goals, which comprise the secrecy of the exchanged session key, are listed below:

Goal 1: $EV \Big|\equiv \left(EV \overset{SK}{\leftrightarrow} EAG\right)$

Goal 2: $EV \mid\equiv EAG \mid \equiv \left(EV \overset{SK}{\leftrightarrow} EAG\right)$

Goal 3: $EAG \Big|\equiv \left(EV \overset{SK}{\leftrightarrow} EAG\right)$

Goal 4: $EAG \mid\equiv EV \mid \equiv \left(EV \overset{SK}{\leftrightarrow} EAG\right)$

Step 2: Assumptions. The proposed protocol's preliminary assumptions are as follows:

P1. $EAG \mid\equiv \# (T_{EV})$

P2. $EV \mid\equiv \# (T_{EAG})$

P3. $EAG \mid\equiv \# (N_{EV})$

P4. $EAG \mid\equiv (EV \overset{N_{EV}, T_{EV}}{\leftrightarrow} EAG)$

P5. $EV \mid\equiv (EV \overset{T_{EAG}}{\leftrightarrow} EAG)$

P6. $EAG \mid\equiv (EV \overset{K}{\leftrightarrow} EAG)$

P7. $EV \mid\equiv (EV \overset{K}{\leftrightarrow} EAG)$

P8. $EAG \mid\equiv EV \mid\Rightarrow (EV \overset{SK}{\leftrightarrow} EAG)$

P9. $EV \mid\equiv EAG \mid\Rightarrow (EV \overset{SK}{\leftrightarrow} EAG)$

Step 3: Idealization. The following is an idealized version of the proposed protocol:

M1. EV → EAG: $\left(\left\{ \rightarrow \underset{\rightarrow}{PK_{EV}}\ EV \right\rangle_{\mathrm{A}_{EV}}, N_{EV}, T_{EV} \right\}_{\underset{\rightarrow}{PK_{EAG}}\ EAG})$

M2. EAG → EV: $(\{\langle AT_{No}\rangle_{AT_{EV}^{EAG}}, T_{EAG}\}_{h(id_{EAG}, N_{EV}, T_{EV})}$, $\{Charge\}_{h(EV \overset{K}{\leftrightarrow} EAG,\, AT_{No},\, \underset{\rightarrow}{PK_{EV}}\ EV,\, N_{EV})})$

Step 4: Analysis. The beliefs that both the EV and EAG can obtain in the proposed protocol are derived here. Then, we investigate, based on BAN logic rules, which authentication goals can be met.

Statement 1: Applying the see rule to M1, we obtain:

$$S1:\ EAG \triangleleft \ \left(\left\{\left\langle \underset{\rightarrow}{PK_{EV}}\ EV \right\rangle_{A_{EV}}, N_{EV}, T_{EV} \right\}_{\underset{\rightarrow}{PK_{EAG}}\ EAG}\right)$$

Statement 2: In accordance with Rule (1) (message-meaning rule) and $S1$ and P6, we obtain:

$$S2:\ EAG \mid\equiv\ EV \mid \sim \left(\left\{\left\langle \underset{\rightarrow}{PK_{EV}}\ EV \right\rangle_{A_{EV}}, N_{EV}, T_{EV} \right\}_{\underset{\rightarrow}{PK_{EAG}}\ EAG}\right)$$

Statement 3: In accordance with freshness conjuncatenation (Rule (4)) and nonce verification (Rule (2)) with $S2$, P1, and P3, we obtain:

$$S3:\ EAG \mid\equiv EV \mid \equiv (\left\{ \left\langle \overset{PK_{EV}}{\rightarrow} EV \right\rangle_{A_{EV}}, N_{EV}, T_{EV} \right\}_{\overset{PK_{EAG}}{\rightarrow} EAG})$$

Statement 4: Since the session key $SK_{EV-EAG} = h\left(EV \overset{K}{\leftrightarrow} EAG,\ AT_{No},\ \overset{PK_{EV}}{\rightarrow} EV,\ N_{EV} \right)$ and based on the session keys rule (Rule (6)) with $S3$ and P4, we obtain:

$$S4:\ EAG \mid\equiv\ EV \mid \equiv \left(EV \overset{SK}{\leftrightarrow}\ EAG \right)$$
(Goal 4)

Statement 5: Based on the jurisdiction (Rule (3)) with $S4$ and P8, we obtain:

$$S5:\ EAG \mid \equiv\ \left(EV \overset{SK}{\leftrightarrow}\ EAG \right)$$
(Goal 3)

Statement 6: Applying the see rule to M2, we obtain:

$$S6:\ EV\ \lhd\ (\{\langle AT_{No}\rangle_{AT_{EV}^{EAG}},\ T_{EAG}\}_{h(id_{EAG},\ N_{EV},\ T_{EV})},\ \{Charge\}_{h(EV \overset{K}{\leftrightarrow} EAG,\ AT_{No},\ \overset{PK_{EV}}{\rightarrow} EV,\ N_{EV})})$$

Statement 7: In accordance with message meaning (Rule (1)) with $S6$ and P7, we obtain:

$$S7:\ EV \mid\equiv\ EAG \mid \sim\ (\{\langle AT_{No}\rangle_{AT_{EV}^{EAG}},\ T_{EAG}\}_{h(id_{EAG},\ N_{EV},\ T_{EV})},\ \{Charge\}_{h(EV \overset{K}{\leftrightarrow} EAG,\ AT_{No},\ \overset{PK_{EV}}{\rightarrow} EV,\ N_{EV})})$$

Statement 8: In accordance with freshness conjuncatenation (Rule (4)) and nonce verification (Rule (2)) with $S7$ and P2, we obtain:

$$S8:\ EV \mid\equiv\ EAG \mid \equiv\ (\{\langle AT_{No}\rangle_{AT_{EV}^{EAG}},\ T_{EAG}\}_{h(id_{EAG},\ N_{EV},\ T_{EV})},\ \{Charge\}_{h(EV \overset{K}{\leftrightarrow} EAG,\ AT_{No},\ \overset{PK_{EV}}{\rightarrow} EV,\ N_{EV})})$$

Statement 9: Since the session key $SK_{EV-EAG} = h\left(EV \overset{K}{\leftrightarrow} EAG,\ AT_{No},\ \overset{PK_{EV}}{\rightarrow} EV,\ N_{EV} \right)$ and based on the session keys rule (Rule (6)) with $S8$ and P5, we obtain:

$$S9:\ EV \mid\equiv\ EAG \mid \equiv \left(EV \overset{SK}{\leftrightarrow}\ EAG \right)$$
(Goal 2)

Statement 10: Based on the jurisdiction (Rule (3)) with $S9$ and P9, we obtain:

$$S10:\ EV \mid \equiv\ \left(EV \overset{SK}{\leftrightarrow}\ EAG \right)$$
(Goal 1)

In summary, the proposed protocol provides the EV and EAG with secure mutual authentication. Additionally, based on the achieved Goals 1, 2, 3, and 4, the EV and EAG can confidently share the session key (SK). Accordingly, using BAN logic, we could say that the proposed protocol provides secure mutual authentication, ensuring that the EV and EAG are the only parties with access to the session key, maintaining security.

5.1.2. Analyzing Reauthentication Protocol

To prove that the proposed reauthentication protocol is accurate, the steps listed below are performed.

Step 1: Goals. The analysis' key goals, which comprise the secrecy of the exchanged session key, are listed below:

Goal 1: $EV \left| \equiv \left(EV \overset{SK'}{\leftrightarrow} EAG \right) \right.$

Goal 2: $EV \mid\equiv EAG \mid \equiv \left(EV \overset{SK'}{\leftrightarrow} EAG \right)$

Goal 3: $EAG \left| \equiv \left(EV \overset{SK'}{\leftrightarrow} EAG \right) \right.$

Goal 4: $EAG \mid\equiv EV \mid \equiv \left(EV \overset{SK'}{\leftrightarrow} EAG \right)$

Step 2: Assumptions. The proposed protocol's preliminary assumptions are as follows:

P1. $EAG \mid\equiv \# (N'_{EV})$
P2. $EAG \mid\equiv \# (T_{EV})$
P3. $EAG \mid\equiv \# (N_{EV})$
P4. $EV \mid\equiv \# (N_{EAG})$
P5. $EAG \mid\equiv (EV \overset{N_{EV}, N'_{EV}, T_{EV}}{\leftrightarrow} EAG)$
P6. $EV \mid\equiv (EV \overset{N_{EAG}}{\leftrightarrow} EAG)$
P7. $EAG \mid\equiv (EV \overset{K}{\leftrightarrow} EAG)$
P8. $EV \mid\equiv (EV \overset{K}{\leftrightarrow} EAG)$
P9. $EAG \mid\equiv EV \mid\Rightarrow (EV \overset{SK'}{\leftrightarrow} EAG)$
P10. $EV \mid\equiv EAG \mid\Rightarrow (EV \overset{SK'}{\leftrightarrow} EAG)$

Step 3: Idealization. The following is an idealized version of the proposed protocol:

M1. EV → EAG: $((\langle AT_{No} \rangle_{AT_{EV}^{EAG}}, \left\{ \underset{\rightarrow}{PK_{EV}}\ EV, EV \overset{K}{\leftrightarrow} EAG \right\}_{\underset{\rightarrow}{PK_{EAG}}\ EAG}),$

$N_E, \left\{ N'_{EV},\ T_{EV},\ \langle Aid_{No} \rangle_{\mathbf{A}_{EV}} \right\}_{h(EV \overset{K}{\leftrightarrow} EAG, AT_{No},\ \underset{\rightarrow}{PK_{EV}}\ EV, N_{EV})})$

M2. EAG → EV: $(\{N_{EAG}\}_{h(AT_{No}, Aid_{No}, N'_{EV}, T_{EV})},$

$\{Charge\}_{h(EV \overset{TK}{\leftrightarrow} EAG, EV \overset{K}{\leftrightarrow} EAG, N_{EAG},\ \underset{\rightarrow}{PK_{EV}}\ EV)})$

Step 4: Analysis. The beliefs that both the EV and EAG can obtain in the proposed protocol are derived here. Then, we investigate which authentication goals can be met.

Statement 1: Applying the see rule to M1, we obtain:

$$S1:\ EAG \triangleleft ((\langle AT_{No} \rangle_{AT_{EV}^{EAG}}, \left\{ \underset{\rightarrow}{PK_{EV}}\ EV,\ EV \overset{K}{\leftrightarrow} EAG \right\}_{\underset{\rightarrow}{PK_{EAG}}\ EAG}),$$

$$N_E,\ \left\{ N'_{EV},\ T_{EV},\ \langle Aid_{No} \rangle_{A_{EV}} \right\}_{h(EV \overset{K}{\leftrightarrow} EAG,\ AT_{No},\ \underset{\rightarrow}{PK_{EV}}\ EV,\ N_{EV})})$$

Statement 2: In accordance with Rule (1) (message-meaning rule) and $S1$ and P7, we obtain:

$$S2: \ EAG \mid\equiv \ EV \mid \sim (((\langle AT_{No} \rangle_{AT_{EV}^{EAG}}, \left\{ \overset{PK_{EV}}{\rightarrow} EV, \ EV \overset{K}{\leftrightarrow} EAG \right\}_{\overset{PK_{EAG}}{\rightarrow} EAG}),$$
$$N_E, \left\{ N'_{EV}, \ T_{EV}, \ \langle Aid_{No} \rangle_{\mathbf{A}_{EV}} \right\}_{h(EV \overset{K}{\leftrightarrow} EAG, AT_{No}, \ \overset{PK_{EV}}{\rightarrow} EV, N_{EV})})$$

Statement 3: In accordance with freshness conjuncatenation (Rule (4)) and nonce verification (Rule (2)) with $S2$, P1, P2, and P3, we obtain:

$$S3: \ EAG \mid\equiv EV \mid \equiv (\langle AT_{No} \rangle_{AT_{EV}^{EAG}}, \left\{ \overset{PK_{EV}}{\rightarrow} EV, \ EV \overset{K}{\leftrightarrow} EAG \right\}_{\overset{PK_{EAG}}{\rightarrow} EAG}),$$
$$N_E, \ \left\{ N'_{EV}, \ T_{EV}, \ \langle Aid_{No} \rangle_{A_{EV}} \right\}_{h(EV \overset{K}{\leftrightarrow} EAG, \ AT_{No}, \ \overset{PK_{EV}}{\rightarrow} EV, \ N_{EV})})$$

Statement 4: Since the session key $SK'_{EV-EAG} = h\left(EV \overset{TK}{\leftrightarrow} EAG, EV \overset{K}{\leftrightarrow} EAG, N_{EAG}, \ \overset{PK_{EV}}{\rightarrow} EV \right)$, $TK_{EV-EAG} = h(AT_{No}, Aid_{No}, N'_{EV}, T_{EV})$, and based on the session keys rule (Rule (6)) with $S3$ and P5, we obtain:

$$S4: \ EAG \mid\equiv \ EV \mid \equiv \left(EV \ \overset{SK'}{\leftrightarrow} \ \ EAG \right)$$

(Goal 4)

Statement 5: Based on the jurisdiction (Rule (3)) with $S4$ and P9, we obtain:

$$S5: \ EAG \mid \equiv \ \left(EV \ \overset{SK'}{\leftrightarrow} \ \ EAG \right)$$

(Goal 3)

Statement 6: Applying the see rule to M2, we obtain:

$$S6: \ EV \ \triangleleft \ (\{N_{EAG}\}_{h(AT_{No}, \ Aid_{No}, \ N'_{EV}, \ T_{EV})},$$
$$\{Charge\}_{h(EV \overset{TK}{\leftrightarrow} EAG, \ EV \overset{K}{\leftrightarrow} EAG, \ N_{EAG}, \ \overset{PK_{EV}}{\rightarrow} EV)})$$

Statement 7: In accordance with message meaning (Rule (1)) with $S6$ and P8, we obtain:

$$S7: \ EV \mid\equiv \ EAG \mid \sim (\{N_{EAG}\}_{h(AT_{No}, \ Aid_{No}, \ N'_{EV}, \ T_{EV})},$$
$$\{Charge\}_{h(EV \overset{TK}{\leftrightarrow} EAG, \ EV \overset{K}{\leftrightarrow} EAG, \ N_{EAG}, \ \overset{PK_{EV}}{\rightarrow} EV)})$$

Statement 8: In accordance with freshness conjuncatenation (Rule (4)) and nonce verification (Rule (2)) with $S7$ and P4, we obtain:

$$S8: \ EV \mid\equiv \ EAG \mid \equiv (\{N_{EAG}\}_{h(AT_{No}, \ Aid_{No}, \ N'_{EV}, \ T_{EV})},$$
$$\{Charge\}_{h(EV \overset{TK}{\leftrightarrow} EAG, \ EV \overset{K}{\leftrightarrow} EAG, \ N_{EAG}, \ \overset{PK_{EV}}{\rightarrow} EV)})$$

Statement 9: Since the session key $SK'_{EV-EAG}=h\left(EV \overset{TK}{\leftrightarrow} EAG, EV \overset{K}{\leftrightarrow} EAG, N_{EAG}, \overset{PK_{EV}}{\rightarrow} EV\right)$ and based on the session keys rule (Rule (6)) with $S3$ and P6, we obtain:

$$S9: \; EV \mid\equiv \; EAG \mid \equiv \; \left(EV \overset{SK'}{\leftrightarrow} \; EAG\right)$$

(Goal 2)

Statement 10: Based on the jurisdiction (Rule (3)) with $S9$ and P10, we obtain:

$$S10: \; EV \mid \equiv \; \left(EV \overset{SK'}{\leftrightarrow} \; EAG\right)$$

(Goal 1)

Achieving Goals 1, 2, 3, and 4 implies that the proposed protocol offers secure mutual authentication and the session key is exclusively shared for security between the EV and EAG.

5.2. Security Simulation with AVISPA Tool

A popular formal security verification method used to evaluate if systems or protocols can withstand replay and MITM assaults [46–49] is the automated validation of internet security-sensitive protocols and applications (AVISPA) tool [50], using a security protocol animator (SPAN) [51]. In order to evaluate the authentication protocol's resilience to MITM and replay attacks, we conducted a formal security test of the proposed system using the AVISPA simulation tool.

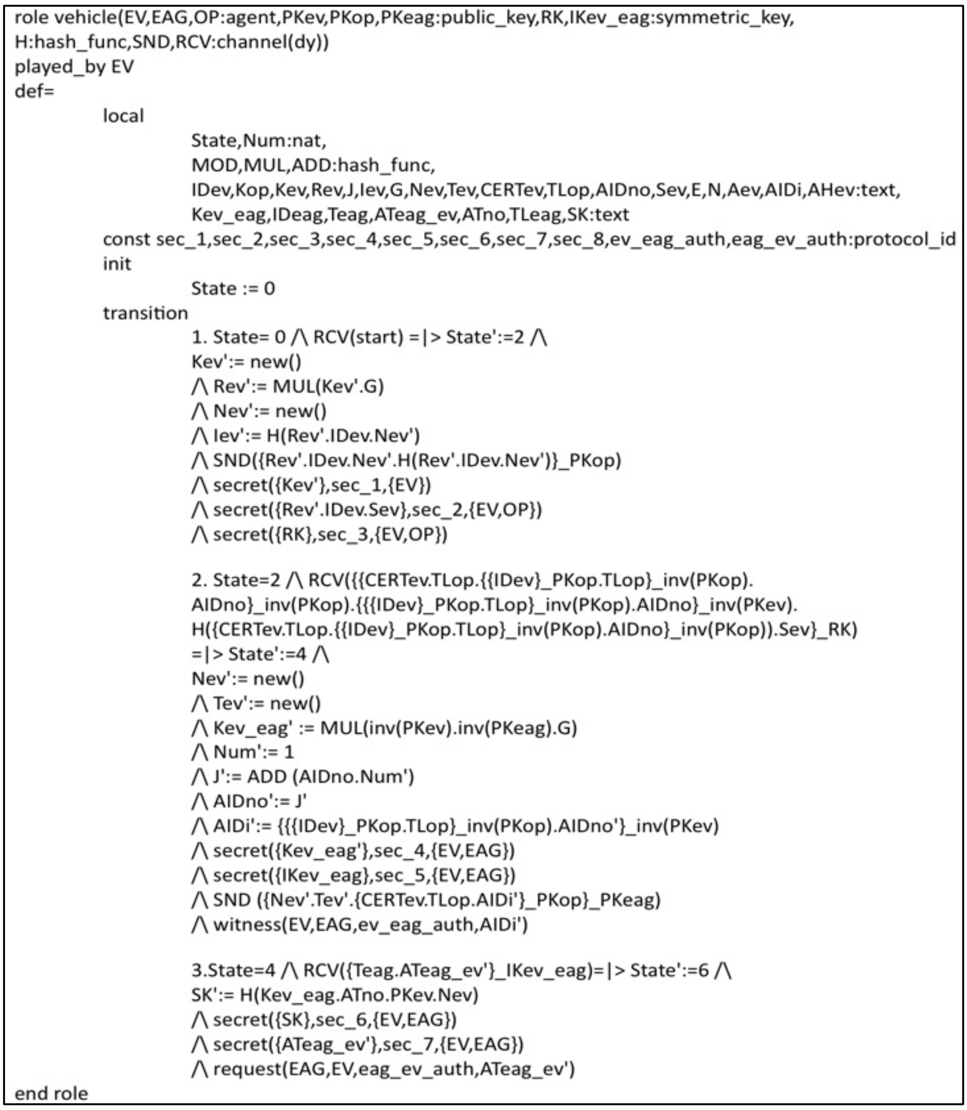
The high-level protocol specification language (HLPSL) was used to write the AVISPA module [52]. The HLPSL is composed of four backends: SAT-based model checker (SATMC), tree, automate-based protocol analyzer (TA4SP), CL-based attack searcher (CL-AtSe) [53], and on-the-fly model checker (OFMC) [54].

5.2.1. Mutual Authentication HLPSL Specification of AVISPA Simulation

Role, session, and environment are the three components of the HLPSL, where role denotes an entity, session denotes system parameters, and environment denotes the knowledge of the intruder, security, and authentication objectives. The mutual authentication HLPSL specifications for different roles (EV, OP, and EAG) are shown in Figures 8–10, respectively. Figure 11 shows the specifications of the session and environment.

The role of the EV is presented in Figure 8. As soon as an EV enters its initial transition (State 0), it obtains the starting request and begins the registration procedure by sending a request $\{id_{EV},\ R_{EV},\ N_{EV},\ I_{EV}\}$ encrypted with PK_{OP} to the OP via an open channel, changes the state value to 2, and then employs the secret function to determine whether the entity is a legitimate user.

In State 2, the EV receives its credentials $\{A_{EV},\ AH_{EV},\ Aid_i, S_{EV}\}$ encrypted with RK from the OP and sends the authentication request $\{A_{EV},\ N_{EV},\ T_{EV}\}$ encrypted with PK_{EAG} to the EAG via an open channel and modifies the state value to 4. Moreover, witness(EV, EAG, ev_eag_auth, Aid'_i) is declared by the EV to show that Aid'_i is a weak authentication factor. In State 4, the EV receives the response $\{AT^{EAG}_{EV},\ T_{EAG}\}$ encrypted by IK_{EV-EAG} from the EAG, modifies the state value to 6, and calculates the session key SK_{EV-EAG}; EV declare *request*(EAG, EV, eag_ev_auth, $AT^{EAG'}_{EV}$) to authenticate each other.



```
role vehicle(EV,EAG,OP:agent,PKev,PKop,PKeag:public_key,RK,IKev_eag:symmetric_key,
H:hash_func,SND,RCV:channel(dy))
played_by EV
def=
          local
                    State,Num:nat,
                    MOD,MUL,ADD:hash_func,
                    IDev,Kop,Kev,Rev,J,Iev,G,Nev,Tev,CERTev,TLop,AIDno,Sev,E,N,Aev,AIDi,AHev:text,
                    Kev_eag,IDeag,Teag,ATeag_ev,ATno,TLeag,SK:text
          const sec_1,sec_2,sec_3,sec_4,sec_5,sec_6,sec_7,sec_8,ev_eag_auth,eag_ev_auth:protocol_id
          init
                    State := 0
          transition
                    1. State= 0 /\ RCV(start) =|> State':=2 /\
                    Kev':= new()
                    /\ Rev':= MUL(Kev'.G)
                    /\ Nev':= new()
                    /\ Iev':= H(Rev'.IDev.Nev')
                    /\ SND({Rev'.IDev.Nev'.H(Rev'.IDev.Nev')}_PKop)
                    /\ secret({Kev'},sec_1,{EV})
                    /\ secret({Rev'.IDev.Sev},sec_2,{EV,OP})
                    /\ secret({RK},sec_3,{EV,OP})

                    2. State=2 /\ RCV({{CERTev.TLop.{{IDev}_PKop.TLop}_inv(PKop).
                    AIDno}_inv(PKop).{{{IDev}_PKop.TLop}_inv(PKop).AIDno}_inv(PKev).
                    H({CERTev.TLop.{{IDev}_PKop.TLop}_inv(PKop).AIDno}_inv(PKop)).Sev}_RK)
                    =|> State':=4 /\
                    Nev':= new()
                    /\ Tev':= new()
                    /\ Kev_eag' := MUL(inv(PKev).inv(PKeag).G)
                    /\ Num':= 1
                    /\ J':= ADD (AIDno.Num')
                    /\ AIDno':= J'
                    /\ AIDi':= {{{IDev}_PKop.TLop}_inv(PKop).AIDno'}_inv(PKev)
                    /\ secret({Kev_eag'},sec_4,{EV,EAG})
                    /\ secret({IKev_eag},sec_5,{EV,EAG})
                    /\ SND ({Nev'.Tev'.{CERTev.TLop.AIDi'}_PKop}_PKeag)
                    /\ witness(EV,EAG,ev_eag_auth,AIDi')

                    3.State=4 /\ RCV({Teag.ATeag_ev'}_IKev_eag)=|> State':=6 /\
                    SK':= H(Kev_eag.ATno.PKev.Nev)
                    /\ secret({SK},sec_6,{EV,EAG})
                    /\ secret({ATeag_ev'},sec_7,{EV,EAG})
                    /\ request(EAG,EV,eag_ev_auth,ATeag_ev')
end role
```

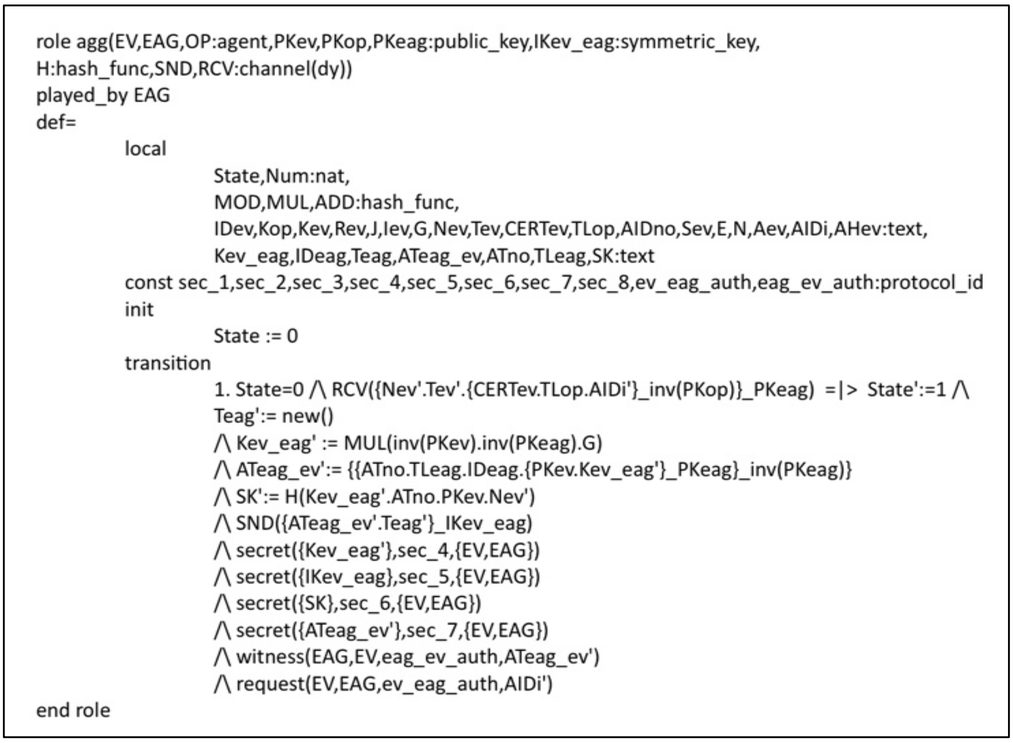
Figure 8. Mutual Authentication Specification of EV.

5.2.2. Mutual Authentication AVISPA Verification Results

In order to assess the proposed mutual authentication scheme's security, we display the AVISPA findings and use the OFMC and CL-AtSe. The OFMC validates that the proposed system is safe against MITM attacks. Furthermore, the CL-AtSe illustrates the protocol's resistance to replay attacks. The proposed mutual authentication technique is safe against MITM and replay attacks, as shown in Figure 12, which shows the results of the AVISPA simulation.

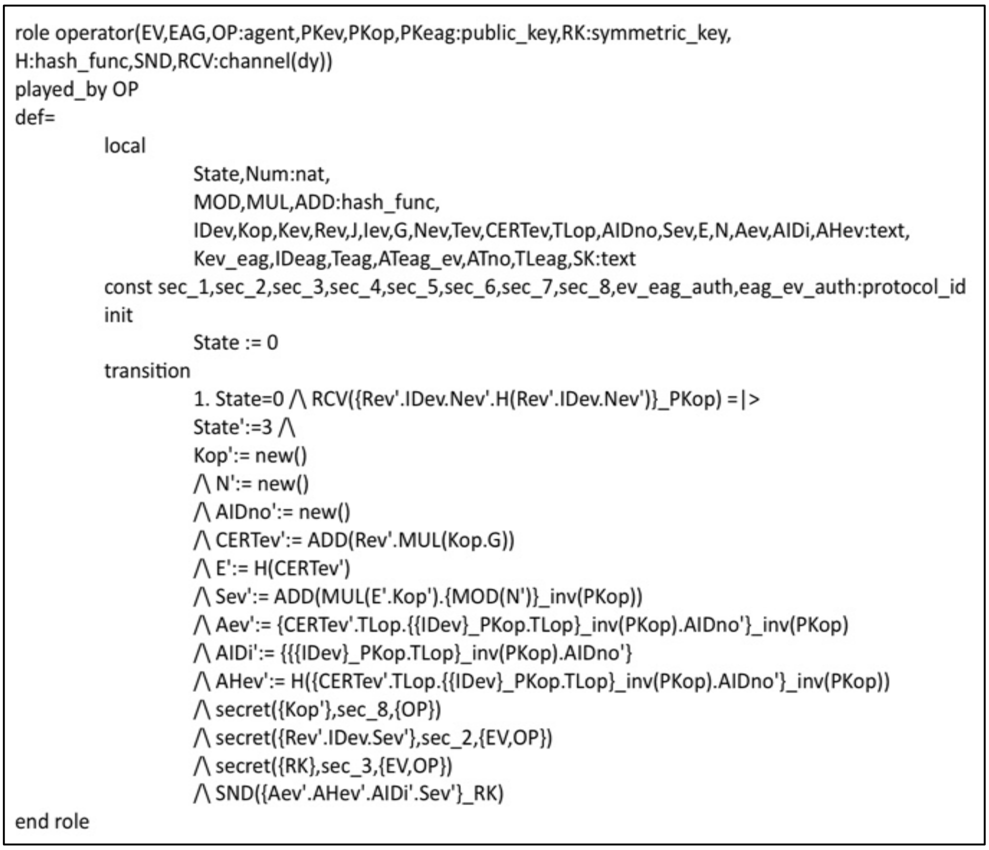
5.2.3. Reauthentication HLPSL Specifications of AVISPA Simulation

The reauthentication HLPSL specifications for different roles (EV and EAG) are shown in Figures 13 and 14, respectively. Figure 15 shows the specifications of the session and environment.



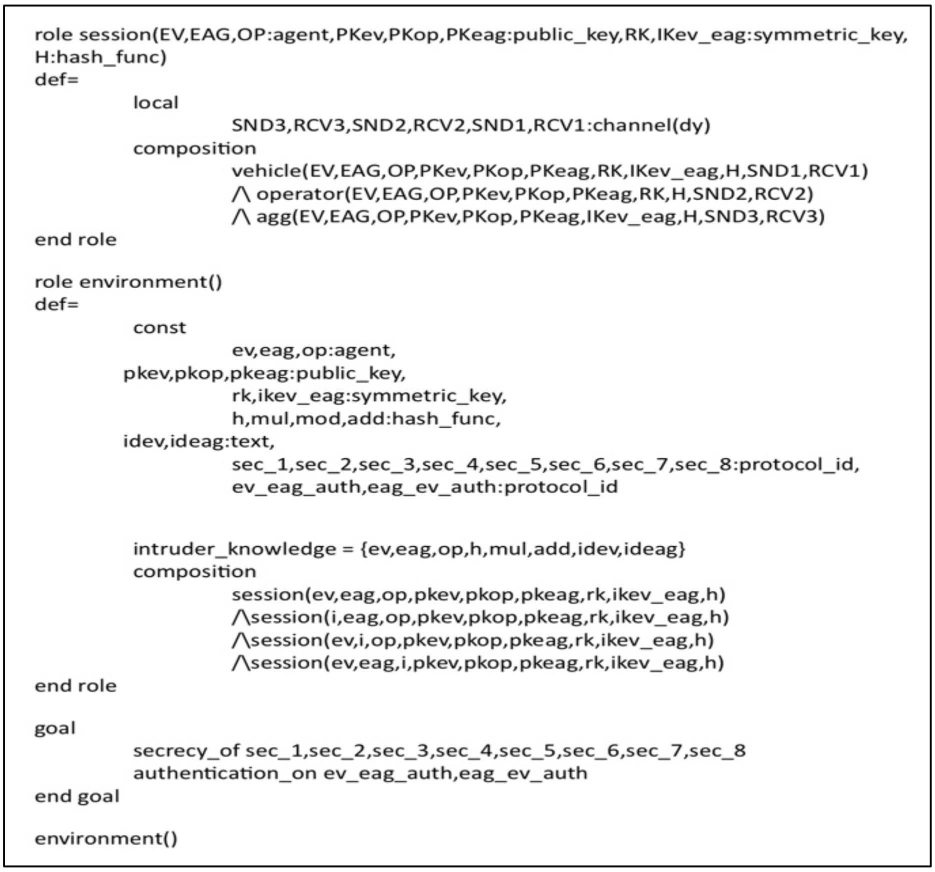
```
role agg(EV,EAG,OP:agent,PKev,PKop,PKeag:public_key,IKev_eag:symmetric_key,
H:hash_func,SND,RCV:channel(dy))
played_by EAG
def=
        local
                State,Num:nat,
                MOD,MUL,ADD:hash_func,
                IDev,Kop,Kev,Rev,J,Iev,G,Nev,Tev,CERTev,TLop,AIDno,Sev,E,N,Aev,AIDi,AHev:text,
                Kev_eag,IDeag,Teag,ATeag_ev,ATno,TLeag,SK:text
        const sec_1,sec_2,sec_3,sec_4,sec_5,sec_6,sec_7,sec_8,ev_eag_auth,eag_ev_auth:protocol_id
        init
                State := 0
        transition
                1. State=0 /\ RCV({Nev'.Tev'.{CERTev.TLop.AIDi'}_inv(PKop)}_PKeag)  =|> State':=1 /\
                Teag':= new()
                /\ Kev_eag' := MUL(inv(PKev).inv(PKeag).G)
                /\ ATeag_ev':= {{ATno.TLeag.IDeag.{PKev.Kev_eag'}_PKeag}_inv(PKeag)}
                /\ SK':= H(Kev_eag'.ATno.PKev.Nev')
                /\ SND({ATeag_ev'.Teag'}_IKev_eag)
                /\ secret({Kev_eag'},sec_4,{EV,EAG})
                /\ secret({IKev_eag},sec_5,{EV,EAG})
                /\ secret({SK},sec_6,{EV,EAG})
                /\ secret({ATeag_ev'},sec_7,{EV,EAG})
                /\ witness(EAG,EV,eag_ev_auth,ATeag_ev')
                /\ request(EV,EAG,ev_eag_auth,AIDi')
end role
```

Figure 9. Mutual Authentication Specification of EAG.



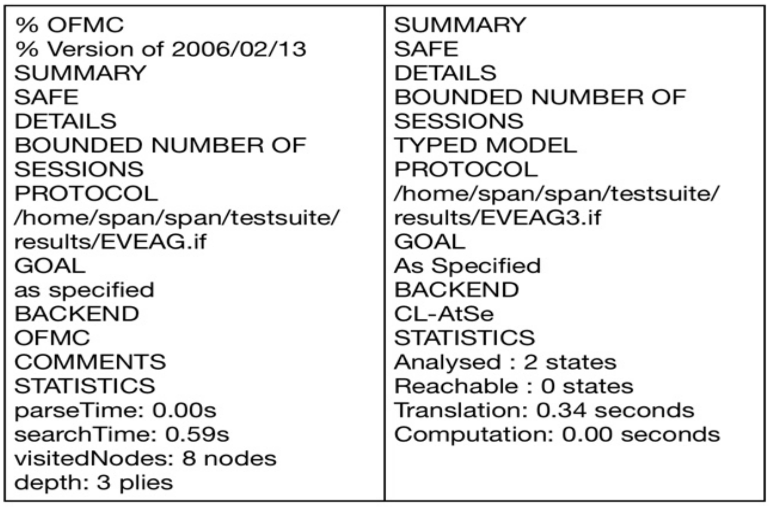
```
role operator(EV,EAG,OP:agent,PKev,PKop,PKeag:public_key,RK:symmetric_key,
H:hash_func,SND,RCV:channel(dy))
played_by OP
def=
        local
                State,Num:nat,
                MOD,MUL,ADD:hash_func,
                IDev,Kop,Kev,Rev,J,Iev,G,Nev,Tev,CERTev,TLop,AIDno,Sev,E,N,Aev,AIDi,AHev:text,
                Kev_eag,IDeag,Teag,ATeag_ev,ATno,TLeag,SK:text
        const sec_1,sec_2,sec_3,sec_4,sec_5,sec_6,sec_7,sec_8,ev_eag_auth,eag_ev_auth:protocol_id
        init
                State := 0
        transition
                1. State=0 /\ RCV({Rev'.IDev.Nev'.H(Rev'.IDev.Nev')}_PKop) =|>
                State':=3 /\
                Kop':= new()
                /\ N':= new()
                /\ AIDno':= new()
                /\ CERTev':= ADD(Rev'.MUL(Kop.G))
                /\ E':= H(CERTev')
                /\ Sev':= ADD(MUL(E'.Kop').{MOD(N')}_inv(PKop))
                /\ Aev':= {CERTev'.TLop.{{IDev}_PKop.TLop}_inv(PKop).AIDno'}_inv(PKop)
                /\ AIDi':= {{{IDev}_PKop.TLop}_inv(PKop).AIDno'}
                /\ AHev':= H({CERTev'.TLop.{{IDev}_PKop.TLop}_inv(PKop).AIDno'}_inv(PKop))
                /\ secret({Kop'},sec_8,{OP})
                /\ secret({Rev'.IDev.Sev'},sec_2,{EV,OP})
                /\ secret({RK},sec_3,{EV,OP})
                /\ SND({Aev'.AHev'.AIDi'.Sev'}_RK)
end role
```

Figure 10. Mutual Authentication Specification of OP.



```
role session(EV,EAG,OP:agent,PKev,PKop,PKeag:public_key,RK,IKev_eag:symmetric_key,
H:hash_func)
def=
          local
                    SND3,RCV3,SND2,RCV2,SND1,RCV1:channel(dy)
          composition
                    vehicle(EV,EAG,OP,PKev,PKop,PKeag,RK,IKev_eag,H,SND1,RCV1)
                    /\ operator(EV,EAG,OP,PKev,PKop,PKeag,RK,H,SND2,RCV2)
                    /\ agg(EV,EAG,OP,PKev,PKop,PKeag,IKev_eag,H,SND3,RCV3)
end role

role environment()
def=
          const
                    ev,eag,op:agent,
         pkev,pkop,pkeag:public_key,
                    rk,ikev_eag:symmetric_key,
                    h,mul,mod,add:hash_func,
         idev,ideag:text,
                    sec_1,sec_2,sec_3,sec_4,sec_5,sec_6,sec_7,sec_8:protocol_id,
                    ev_eag_auth,eag_ev_auth:protocol_id


          intruder_knowledge = {ev,eag,op,h,mul,add,idev,ideag}
          composition
                    session(ev,eag,op,pkev,pkop,pkeag,rk,ikev_eag,h)
                    /\session(i,eag,op,pkev,pkop,pkeag,rk,ikev_eag,h)
                    /\session(ev,i,op,pkev,pkop,pkeag,rk,ikev_eag,h)
                    /\session(ev,eag,i,pkev,pkop,pkeag,rk,ikev_eag,h)
end role

goal
          secrecy_of sec_1,sec_2,sec_3,sec_4,sec_5,sec_6,sec_7,sec_8
          authentication_on ev_eag_auth,eag_ev_auth
end goal

environment()
```

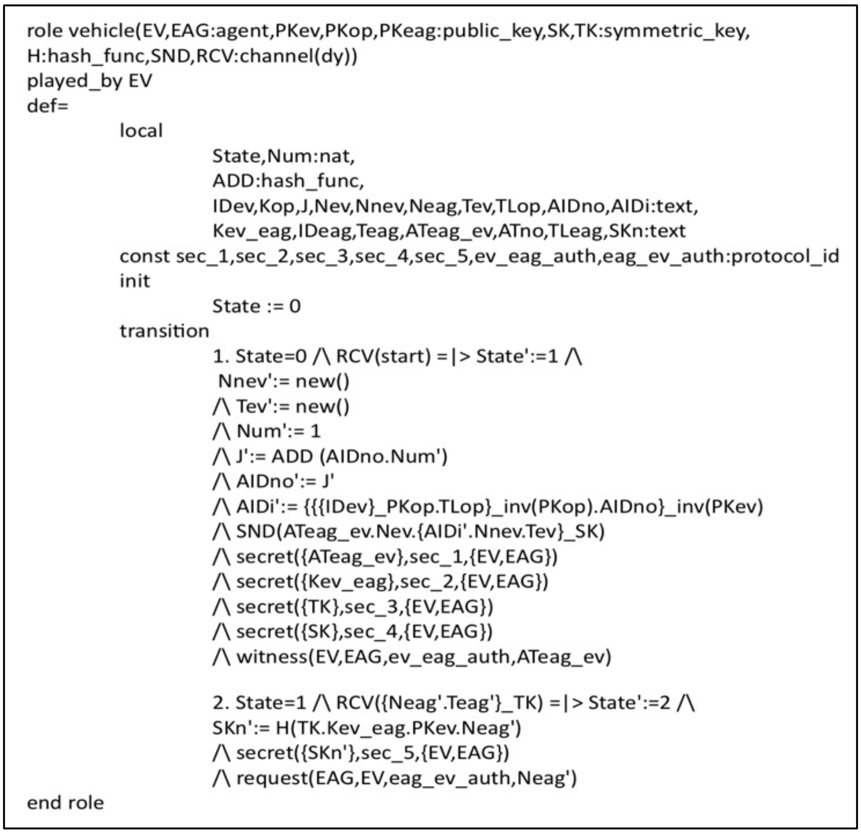
Figure 11. Mutual Authentication Specification of the session and environment.



```
% OFMC
% Version of 2006/02/13
SUMMARY
SAFE
DETAILS
BOUNDED NUMBER OF
SESSIONS
PROTOCOL
/home/span/span/testsuite/
results/EVEAG.if
GOAL
as specified
BACKEND
OFMC
COMMENTS
STATISTICS
parseTime: 0.00s
searchTime: 0.59s
visitedNodes: 8 nodes
depth: 3 plies
```

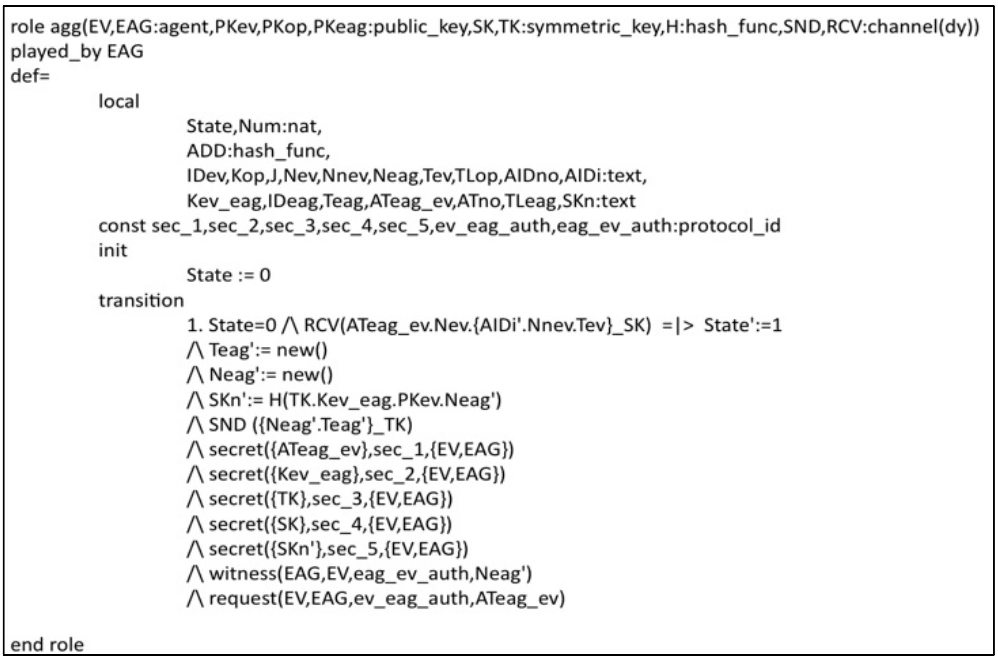
```
SUMMARY
SAFE
DETAILS
BOUNDED NUMBER OF
SESSIONS
TYPED MODEL
PROTOCOL
/home/span/span/testsuite/
results/EVEAG3.if
GOAL
As Specified
BACKEND
CL-AtSe
STATISTICS
Analysed : 2 states
Reachable : 0 states
Translation: 0.34 seconds
Computation: 0.00 seconds
```

Figure 12. Mutual Authentication: AVISPA simulation results.



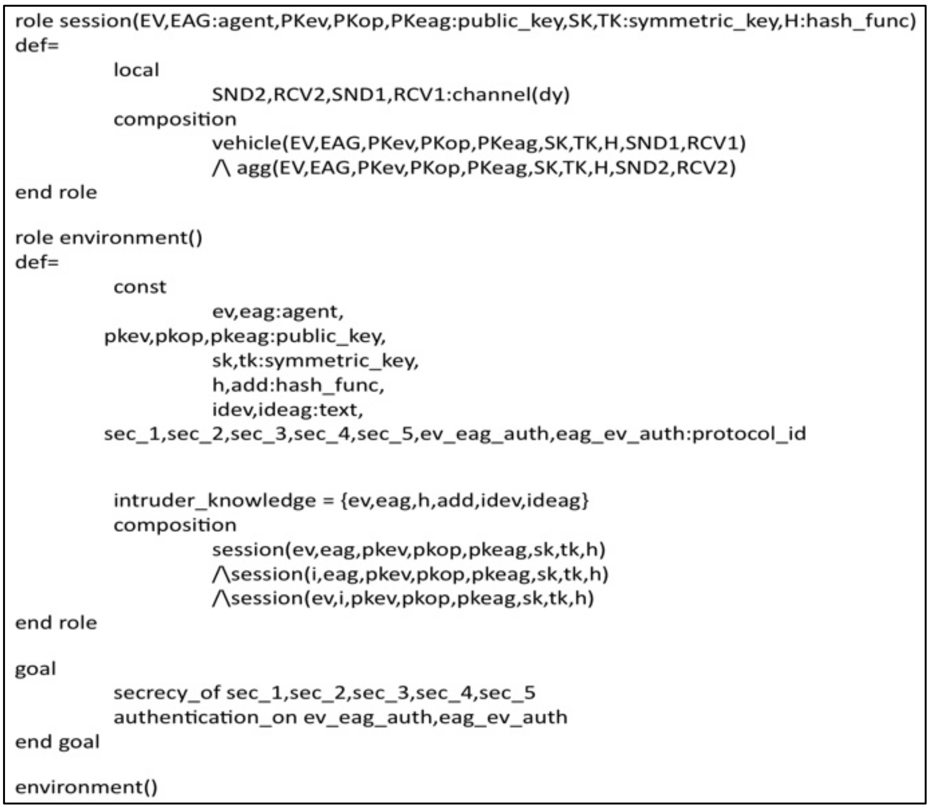
```
role vehicle(EV,EAG:agent,PKev,PKop,PKeag:public_key,SK,TK:symmetric_key,
H:hash_func,SND,RCV:channel(dy))
played_by EV
def=
        local
                State,Num:nat,
                ADD:hash_func,
                IDev,Kop,J,Nev,Nnev,Neag,Tev,TLop,AIDno,AIDi:text,
                Kev_eag,IDeag,Teag,ATeag_ev,ATno,TLeag,SKn:text
        const sec_1,sec_2,sec_3,sec_4,sec_5,ev_eag_auth,eag_ev_auth:protocol_id
        init
                State := 0
        transition
                1. State=0 /\ RCV(start) =|> State':=1 /\
                 Nnev':= new()
                /\ Tev':= new()
                /\ Num':= 1
                /\ J':= ADD (AIDno.Num')
                /\ AIDno':= J'
                /\ AIDi':= {{{IDev}_PKop.TLop}_inv(PKop).AIDno}_inv(PKev)
                /\ SND(ATeag_ev.Nev.{AIDi'.Nnev.Tev}_SK)
                /\ secret({ATeag_ev},sec_1,{EV,EAG})
                /\ secret({Kev_eag},sec_2,{EV,EAG})
                /\ secret({TK},sec_3,{EV,EAG})
                /\ secret({SK},sec_4,{EV,EAG})
                /\ witness(EV,EAG,ev_eag_auth,ATeag_ev)

                2. State=1 /\ RCV({Neag'.Teag'}_TK) =|> State':=2 /\
                SKn':= H(TK.Kev_eag.PKev.Neag')
                /\ secret({SKn'},sec_5,{EV,EAG})
                /\ request(EAG,EV,eag_ev_auth,Neag')
end role
```

Figure 13. Reauthentication Specification of EV.



```
role agg(EV,EAG:agent,PKev,PKop,PKeag:public_key,SK,TK:symmetric_key,H:hash_func,SND,RCV:channel(dy))
played_by EAG
def=
        local
                State,Num:nat,
                ADD:hash_func,
                IDev,Kop,J,Nev,Nnev,Neag,Tev,TLop,AIDno,AIDi:text,
                Kev_eag,IDeag,Teag,ATeag_ev,ATno,TLeag,SKn:text
        const sec_1,sec_2,sec_3,sec_4,sec_5,ev_eag_auth,eag_ev_auth:protocol_id
        init
                State := 0
        transition
                1. State=0 /\ RCV(ATeag_ev.Nev.{AIDi'.Nnev.Tev}_SK)  =|>  State':=1
                /\ Teag':= new()
                /\ Neag':= new()
                /\ SKn':= H(TK.Kev_eag.PKev.Neag')
                /\ SND ({Neag'.Teag'}_TK)
                /\ secret({ATeag_ev},sec_1,{EV,EAG})
                /\ secret({Kev_eag},sec_2,{EV,EAG})
                /\ secret({TK},sec_3,{EV,EAG})
                /\ secret({SK},sec_4,{EV,EAG})
                /\ secret({SKn'},sec_5,{EV,EAG})
                /\ witness(EAG,EV,eag_ev_auth,Neag')
                /\ request(EV,EAG,ev_eag_auth,ATeag_ev)

end role
```

Figure 14. Reauthentication Specification of EAG.



```
role session(EV,EAG:agent,PKev,PKop,PKeag:public_key,SK,TK:symmetric_key,H:hash_func)
def=
        local
                SND2,RCV2,SND1,RCV1:channel(dy)
        composition
                vehicle(EV,EAG,PKev,PKop,PKeag,SK,TK,H,SND1,RCV1)
                /\ agg(EV,EAG,PKev,PKop,PKeag,SK,TK,H,SND2,RCV2)
end role

role environment()
def=
        const
                ev,eag:agent,
       pkev,pkop,pkeag:public_key,
                sk,tk:symmetric_key,
                h,add:hash_func,
                idev,ideag:text,
       sec_1,sec_2,sec_3,sec_4,sec_5,ev_eag_auth,eag_ev_auth:protocol_id


        intruder_knowledge = {ev,eag,h,add,idev,ideag}
        composition
                session(ev,eag,pkev,pkop,pkeag,sk,tk,h)
                /\session(i,eag,pkev,pkop,pkeag,sk,tk,h)
                /\session(ev,i,pkev,pkop,pkeag,sk,tk,h)
end role

goal
        secrecy_of sec_1,sec_2,sec_3,sec_4,sec_5
        authentication_on ev_eag_auth,eag_ev_auth
end goal

environment()
```

Figure 15. Reauthentication Specification of the session and environment.

The role of EV is presented in Figure 13. At an EV's first transition (State 0), it receives the starting request and then the EV sends the request $\{AT_{EV}^{EAG}, N_{EV}, \left\{Aid_i', N_{EV}'.T_{EV}\right\}$ $SK_{EV-EAG}\}$ to the EAG via an open channel, modifies the state value to 1, and then uses the secret function to validate if the entity is a legitimate user. Moreover, witness(EV, EAG, ev_eag_auth, AT_{EV}^{EAG}) is declared by the EV to show that AT_{EV}^{EAG} is a weak authentication factor.

In State 1, the EV receives the response $\{N_{EAG}', T_{EAG}'\}$ encrypted by TK_{EV-EAG} from the EAG, modifies the state value to 2, and calculates the session key SK_{EV-EAG}'; EV declare request(EAG, EV, eag_ev_auth, N_{EAG}') to authenticate each other.

5.2.4. Reauthentication AVISPA Verification Results

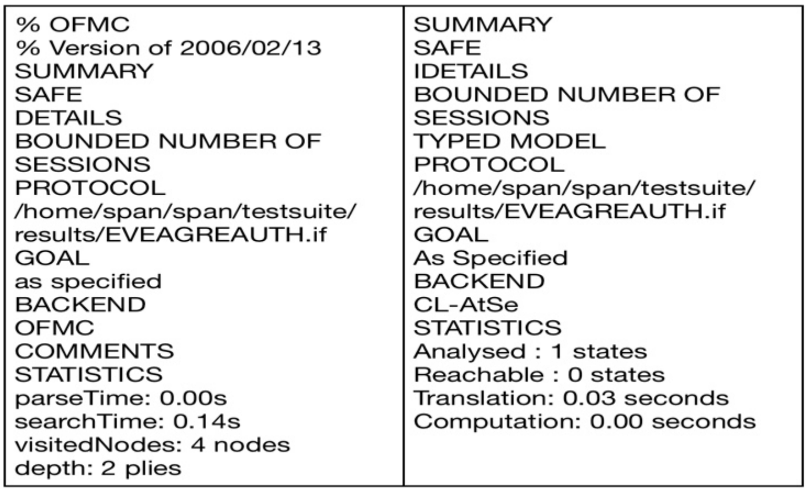
In order to assess the proposed reauthentication scheme's security, we display the AVISPA findings and use the OFMC and CL-AtSe. The proposed reauthentication system is resilient to MITM and replay attacks as an outcome of the AVISPA simulation, as illustrated in Figure 16.

5.3. Informal Security Analysis

To show that the previously stated solution requirements are met, the proposed protocol is analyzed:

5.3.1. Mutual Authentication

Through the verification of the OP's signature on the A_{EAG}, which holds the EAG's identity and certificate $Cert_{EAG}$, the EV can ensure that it interacts with the valid EAG during the authentication phase. The EAG also can validate an EV by two credentials, the OP's signature on the A_{EV}, which contains the EV's certificate $Cert_{EV}$, and the EV's signature Sig_{EV} in its Aid_i. Our system can successfully create mutual authentication among the communicating entities (EV and EAG), as shown by a BAN logic demonstration that we carried out as well.



```
% OFMC
% Version of 2006/02/13
SUMMARY
SAFE
DETAILS
BOUNDED NUMBER OF
SESSIONS
PROTOCOL
/home/span/span/testsuite/
results/EVEAGREAUTH.if
GOAL
as specified
BACKEND
OFMC
COMMENTS
STATISTICS
parseTime: 0.00s
searchTime: 0.14s
visitedNodes: 4 nodes
depth: 2 plies
```

```
SUMMARY
SAFE
IDETAILS
BOUNDED NUMBER OF
SESSIONS
TYPED MODEL
PROTOCOL
/home/span/span/testsuite/
results/EVEAGREAUTH.if
GOAL
As Specified
BACKEND
CL-AtSe
STATISTICS
Analysed : 1 states
Reachable : 0 states
Translation: 0.03 seconds
Computation: 0.00 seconds
```

Figure 16. Reauthentication AVISPA simulation results.

5.3.2. Anonymity

The EV's true identity id_{EV} is encrypted using the OP's public key $PK_{OP}(id_{EV})$ in the Aid_i, which is issued during the registration phase. It can be accessed by the OP only and no other party. Therefore, neither the EAG nor any other party may reveal id_{EV}. This was dissimilar to the studies [26,38], which used the real identity of the EV in the authentication phase, threatening its privacy.

5.3.3. Un-Linkability

For every session, the EV will have an anonymous identity (Aid_i). As the Aid_i was initialized by the OP, EV increments the previous Aid_i sequentially (add 1) for each charging session requested by EV. This was dissimilar to the studies [35,36], where every new charging request contains information about the previous charge, or the EV is associated with a single group until it requests a group change. These techniques enable the adversary to track the targeted EV and threaten its privacy. However, because they have different identities, the proposed protocol sessions are un-linkable to one another.

5.3.4. Traceability

The EV utilizes an anonymous identity Aid_i issued by the OP and even though the EV's real identity is hidden and protected $PK_{OP}(id_{EV})$ within Aid_i, malicious EVs or misbehaving EVs cannot get away unknown by the authorities. Only the OP can reveal id_{EV} if necessary to maintain order in the system. Other schemes such as [27] concealed the real identity even from the system operator to preserve anonymity. Anonymity maintenance should not be absolute in order to protect the system and work as required by all parties.

5.3.5. Forward/Backward Security

The future/old session keys' (SK_{EV-EAG}) secrecy should not be affected in the case an adversary were able to capture any SK_{EV-EAG}. Dynamic session keys are generated for the authentication phase as it includes a unique random number N_{EV}. For the reauthentication phase, it includes a unique random number N_{EAG}. A symmetric temporary key (TK_{EV-EAG}) protects the N_{EAG}, and in every new session TK_{EV-EAG} updates dynamically with a new unique random number N'_{EV}. Moreover, since the master shared key K_{EV-EAG} is a long-lasting key, it was not used to encrypt any message transmitted between the EV and EAG. If the adversary guesses one of the session keys, he/she will only be able to view the relevant communication since each session creates a different session key. As a result, the proposed protocol ensures both forward and backward protection.

5.3.6. Joint Key Control

Without the assistance of other parties (even an OP), a random number N is generated by both parties (EV and EAG) included within the session key (SK_{EV-EAG}) as well as a fresh session key (SK'_{EV-EAG}). Only these two individuals are able to receive the master key K_{EV-EAG} because it is generated by the combination of their private and public keys. Thus, the proposed protocol provides joint key control. This is dissimilar to studies [25,31], where the session key is provided by the service provider.

5.3.7. Effective Reauthentication

To shorten the time consumed and reduce the cost, the EAG reauthenticates the EV within the 48-h time-life of the issued AT_{EV}^{EAG}. The EAG first relies on information from the OP (trusted third party) to authenticate the EV. Afterward, the EAG can authenticate the EV directly without the OP's information, as an EV holds AT_{EV}^{EAG}, unlike the schemes [25–38]. Hence, they established full trust between them (EAG and EV).

5.3.8. Revocation Functionality

To prevent the misuse of stolen tokens (A_{EV}, Aid_i, or AT_{EV}^{EAG}) by an adversary or in the case the EV reports to the OP to revoke its Aid_i A_{EV}, or the EV report to the EAG to revoke its AT_{EV}^{EAG}, the tokens will be considered revoked. The revocation protocol gives the EV a way to alert either the OP or EAG if it suspects an Aid_i, A_{EV}, or AT_{EV}^{EAG} has been stolen through recency proof (RP). Related work [25] did not provide a revocation method for the pseudonyms issued by the service provider.

5.3.9. Resist MITM/Replay Attack

Even if the attacker manages to steal A_{EV}, the master key (K_{EV-EAG}) for EAG authentication must be generated using the EV's private key. In the case that the adversary is able to capture AT_{EV}^{EAG}, the master key (K_{EV-EAG}) cannot be recovered since it is encrypted within AT_{EV}^{EAG} using the EAG's public key. The adversary will, therefore, be unable to generate the session key required for a MITM attack without K_{EV-EAG}. Additionally, in order to prevent replays of previous sessions, random numbers and time stamps are transmitted in every communication between the parties. Therefore, a replay attack is not possible in the proposed protocol.

5.3.10. Resist Impersonation Attack

An adversary with an EAG or OP name cannot generate A_{EV}, Aid_i, or AT_{EV}^{EAG} tokens, as they involve the issuer's signature. So, A_{EV}, Aid_i , and AT_{EV}^{EAG} counterfeiting or cloning is doubtful, because the issuer's signature may be used to verify the legitimacy.

5.3.11. Resist DOS Attack

We incorporated the message-specific puzzles and client puzzles of [55] into our authentication mechanisms to thwart DoS attacks. Each EAG handles access requests normally, that is, without discrimination, when there is no evidence of a DoS attack. However, if an EAG suspects a DoS attack, it selectively executes expensive access request authentication. Particularly, the EAG inserts a special puzzle within the beacon messages and demands the puzzle answer be included in each access request message. Only when the answer is correct does the EAG dedicate resources to handle an access request.

6. Comparison with Related Schemes

The proposed approach is compared with related systems in this section based on security and functional properties as well as computing costs. These are the existing EV charging system schemes that have an emphasis on secure EV-to-EAG communication.

6.1. Security and Functional Features Comparison

Table 4 displays the suggested system's security and functional feature evaluation along with the related systems. There are well-known attacks that could be used against the solutions in [26,30,31,34,38]. Furthermore, solutions in [25–38] do not meet the necessary security and privacy-preservation criteria for EV charging, incorporating traceability, un-linkability, backward security, and efficient re-authentication. In [31,32] it requires an additional message, which increases the overhead on the communication channel. When compared to similar studies, it is shown that our proposed scheme meets all solution requirements.

Table 4. Comparison of the proposed protocol's security features with similar studies. Note: "✓" means "available"; "×" means "not available".

Feature/Approach	**Li et al. [25]**	**Rabieh and Wei [30]**	**Gunukula et al. [31]**	**Huang et al. [26]**	**Roman et al. [36]**	**Kim et al. [27]**	**Roman and Gondim [32]**	**Vaidya and Mouftah [38]**	**Kumar et al. [37]**	**ElGhanam et al. [28]**	**Xia et al. [35]**	**Proposed**
	2016	2017	2017	2018	2019	2019	2019	2020	2020	2021	2021	2022
Mutual Authentication	✓	✓	✓	✓	✓	✓	✓	×	×	✓	✓	✓
Forward security	×	×	×	✓	✓	✓	✓	×	✓	✓	×	✓
Anonymity	✓	✓	✓	×	✓	✓	✓	×	✓	✓	✓	✓
Resist replay attack	✓	✓	×	✓	✓	✓	✓	×	✓	✓	✓	✓
Resist impersonation attack	×	×	×	×	✓	✓	✓	✓	✓	✓	✓	✓
Resist MITM attack	✓	×	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Backward security	×	×	×	×	✓	×	×	×	×	×	×	✓
Un-linkability	×	✓	✓	×	×	×	✓	×	×	✓	×	✓
Traceability	×	×	×	×	×	×	×	×	×	×	✓	✓
Effective Reauthentication	×	×	×	×	×	×	×	×	×	×	×	✓
Revocation method	×	✓	×	×	×	×	✓	×	×	✓	×	✓
Joint key control	×	✓	×	✓	✓	✓	×	×	×	×	×	✓
Number of Messages (EV)	2	2	5	3	2	2	5	3	1	2	1	2

6.2. Computational Cost Comparison

Based on all of the operations that the authentication protocols offer, this section determines the computing cost of the protocols. The EAG performs at a high level; thus, it has the potential to carry out all necessary processes. In contrast to the EAG, the computational resources and memory of EVs are constrained. Therefore, we must focus on the EV computational costs. An illustration of the timing operations is shown in Table 5. Table 6 and Figure 17 indicate the computing costs of the proposed protocol and relevant studies. According to [37,56], a one-way hash function (T_h) takes ≈ 0.0023 milliseconds (ms), the symmetric encryption (T_{sym}) takes ≈ 0.0046 ms, and the elliptic curve encryption ($T_{enc-ecc}$) takes ≈ 0.43 ms.

Table 5. Operations' Timing [37,56].

Notation/Operation	**$T_{enc-ecc}$/Elliptic Curve Encryption**	**T_h/Hash**	**T_{sym}/Symmetric**
Time (ms)	0.43	0.0023	0.0046

Table 6. Comparison of the computational cost.

Approach/Efficiency Feature	EV's Computational Cost	
	Authentication Phase	Reauthentication Phase
Kim et al.'s scheme [27]	$T_{enc-ecc} + 9T_h \approx 0.4484$ ms	$T_{enc-ecc} + 9T_h \approx 0.4484$ ms
Kumar et al.'s scheme [37]	$T_{enc-ecc} + 2T_h \approx 0.4346$ ms	$T_{enc-ecc} + 2T_h \approx 0.4346$ ms
Proposed scheme	$T_{enc-ecc} + 3T_h + 2T_{sym} \approx 0.4461$ ms	$3T_h + 3T_{sym} \approx 0.0207$ ms

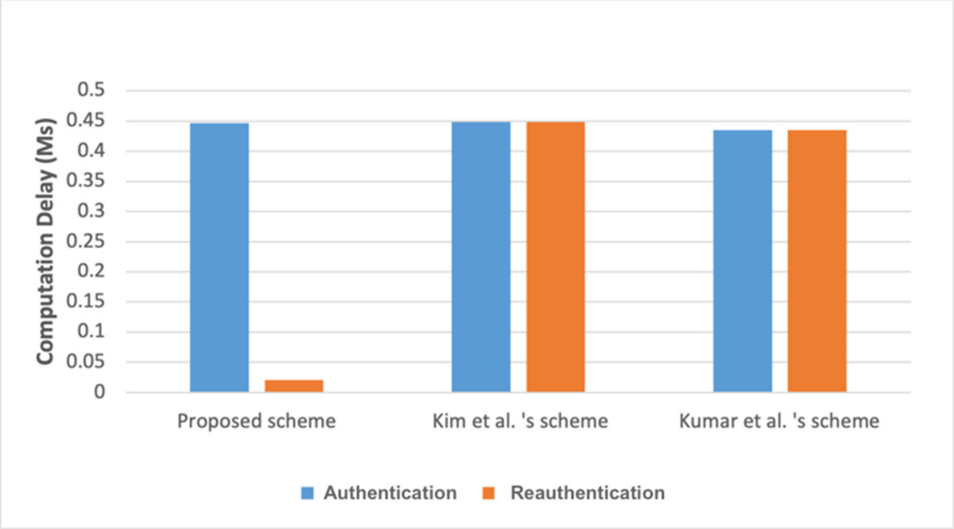


Figure 17. Comparison of several EV's authentication protocols computational cost [27,37].

For the authentication phase, the EV's computational cost in Kim et al.'s scheme [27] requires $\approx$ 0.4484 ms, while in Kumar et al.'s scheme [37], it requires $\approx$ 0.4346 ms. The proposed protocol requires $\approx$ 0.4461 ms; although it is slightly higher than Kumar et al.'s scheme [37], it provides better security and privacy preservation for the EV. In terms of the reauthentication process for the EV, the proposed protocol requires $\approx$ 0.0207 ms, while Kim et al.'s scheme [27] requires $\approx$ 0.4484 ms, and Kumar et al.'s scheme [37] requires $\approx$ 0.4346 ms. Hence, the computational cost of the proposed reauthentication protocol is nearly 95% less than previous protocols. Considering the EV's capability, it is clear now that the proposed protocol outperforms the previous two protocols.

7. Conclusions

An efficient, secure, privacy-preserving authentication system for an electric vehicle charging system is provided in this work. It also includes a reauthentication protocol to minimize the overhead of subsequent authentication processes. A smaller certificate and faster computation have been made available by using the ECQV mechanism's implicit authentication, which is better suited to IoT devices with fewer resources than the conventional certificate. The proposed protocol's ability to perform mutual authentication and meet solution requirements has been demonstrated using BAN logic and informal security analysis. The comparison with previous research reveals that while Kumar et al.'s scheme [37] efficiently reduces the cost of the authentication computational process by around 2.5%, the proposed scheme provides the EV with enhanced security and privacy preservation. However, compared to the other two protocols, the proposed reauthentication protocol outperforms them, with a reduction of about 95%. The real-time experiment is one of the limitations of the proposed scheme. For future work, we intend to study the utilization of machine learning and artificial intelligence to cover a wider range of security. Additionally, we will consider the utilization of a combined certification model (software and hardware certification mechanism). We also suggest studying the possibility of either improving existing evaluation tools (AVISPA, ProVerif, etc.) or exploring new implementations to cover the gap between theoretical analysis and actual implementation.

Author Contributions: Conceptualization, A.M.A. and S.S.A.; methodology, A.M.A. and S.S.A.; validation, A.M.A. and S.S.A.; writing—original draft preparation, S.S.A.; writing—review and editing, A.M.A.; supervision, A.M.A.; funding acquisition, A.M.A. All authors have read and agreed to the published version of the manuscript.

Funding: This work was funded by the SAUDI ARAMCO Cybersecurity Chair at Imam Abdulrahman Bin Faisal University, Saudi Arabia.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: No data were reported in this study.

Acknowledgments: The authors would like to express their appreciation to the Journal Editor, the Associate Editor, and the anonymous reviewers for their insightful comments. We would like to thank Imam Abdulrahman Bin Faisal University for facilitating access to the resources used in this paper.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. US EPA. Sources of Greenhouse Gas Emissions. Available online: https://www.epa.gov/ghgemissions/sources-greenhouse-gas-emissions (accessed on 12 April 2022).
2. Ahmadi, P. Environmental Impacts and Behavioral Drivers of Deep Decarbonization for Transportation through Electric Vehicles. *J. Clean. Prod.* **2019**, *225*, 1209–1219. [CrossRef]
3. Acharya, S.; Dvorkin, Y.; Pandžić, H.; Karri, R. Cybersecurity of Smart Electric Vehicle Charging: A Power Grid Perspective. *IEEE Access* **2020**, *8*, 214434–214453. [CrossRef]
4. Nereim, V. Saudi Arabia to Start Electric-Vehicle Push in Capital Riyadh. Available online: www.bloomberg.com/news/articles/2021-10-23/saudi-arabia-to-start-electric-vehicle-push-in-capital-riyadh (accessed on 23 October 2021).
5. Global EV Outlook 2021—Analysis. IEA. Available online: https://www.iea.org/reports/global-ev-outlook-2021 (accessed on 2 November 2021).
6. Yi, T.; Zhang, C.; Lin, T.; Liu, J. Research on the Spatial-Temporal Distribution of Electric Vehicle Charging Load Demand: A Case Study in China. *J. Clean. Prod.* **2020**, *242*, 118457. [CrossRef]
7. Fu, Z.; Dong, P.; Ju, Y. An Intelligent Electric Vehicle Charging System for New Energy Companies Based on Consortium Blockchain. *J. Clean. Prod.* **2020**, *261*, 121219. [CrossRef]
8. Gorenflo, C.; Golab, L.; Keshav, S. Mitigating Trust Issues in Electric Vehicle Charging Using a Blockchain. In Proceedings of the Tenth ACM International Conference on Future Energy Systems; e-Energy '19. Association for Computing Machinery: New York, NY, USA, 2019; pp. 160–164. [CrossRef]
9. Al-Ogaili, A.S.; Tengku Hashim, T.J.; Rahmat, N.A.; Ramasamy, A.K.; Marsadek, M.B.; Faisal, M.; Hannan, M.A. Review on Scheduling, Clustering, and Forecasting Strategies for Controlling Electric Vehicle Charging: Challenges and Recommendations. *IEEE Access* **2019**, *7*, 128353–128371. [CrossRef]
10. Nedyalkov, I.; Arnaudov, D. Attacks and Security Measures of the Exchanged Information in the Charging Infrastructure for Electromobiles. In Proceedings of the IEEE XXVIII International Scientific Conference Electronics (ET), Sozopol, Bulgaria, 12–14 September 2019; pp. 1–4. [CrossRef]
11. Wang, X.; Hou, X.; Rios, R.; Tippenhauer, N.O.; Ochoa, M. Constrained Proximity Attacks on Mobile Targets. *ACM Trans. Priv. Secur.* **2022**, *25*, 20. [CrossRef]
12. Kilari, V.T.; Yu, R.; Misra, S.; Xue, G. Robust Revocable Anonymous Authentication for Vehicle to Grid Communications. *IEEE Trans. Intell. Transp. Syst.* **2020**, *21*, 4845–4857. [CrossRef]
13. Zhang, X.; Liu, C.; Chai, K.K.; Poslad, S. A Privacy-Preserving Consensus Mechanism for an Electric Vehicle Charging Scheme. *J. Netw. Comput. Appl.* **2021**, *174*, 102908. [CrossRef]
14. ElHussini, H.; Assi, C.; Moussa, B.; Atallah, R.; Ghrayeb, A. A Tale of Two Entities: Contextualizing the Security of Electric Vehicle Charging Stations on the Power Grid. *ACM Trans. Internet Things* **2021**, *2*, 9. [CrossRef]
15. Baroutis, N.; Younis, M. Location Privacy in Wireless Sensor Networks. In *Mission-Oriented Sensor Networks and Systems: Art and Science: Volume 1: Foundations*; Ammari, H.M., Ed.; Studies in Systems, Decision and Control; Springer International Publishing: Cham, Switzerland, 2019; pp. 669–714. [CrossRef]
16. Saxena, N.; Grijalva, S.; Chukwuka, V.; Vasilakos, A.V. Network Security and Privacy Challenges in Smart Vehicle-to-Grid. *IEEE Wirel. Commun.* **2017**, *24*, 88–98. [CrossRef]
17. Hansen, M.; Jensen, M.; Rost, M. Protection Goals for Privacy Engineering. In Proceedings of the 2015 IEEE Security and Privacy Workshops, San Jose, CA, USA, 21–22 May 2015; pp. 159–166. [CrossRef]

18. Mundhe, P.; Verma, S.; Venkatesan, S. A Comprehensive Survey on Authentication and Privacy-Preserving Schemes in VANETs. *Comput. Sci. Rev.* **2021**, *41*, 100411. [CrossRef]
19. Zhang, J.; Cui, J.; Zhong, H.; Chen, Z.; Liu, L. PA-CRT: Chinese Remainder Theorem Based Conditional Privacy-Preserving Authentication Scheme in Vehicular Ad-Hoc Networks. *IEEE Trans. Dependable Secure Comput.* **2021**, *18*, 722–735. [CrossRef]
20. Brown, D.R.L.; Gallant, R.; Vanstone, S.A. Provably Secure Implicit Certificate Schemes. In *Financial Cryptography*; Syverson, P., Ed.; Lecture Notes in Computer Science; Springer: Berlin/Heidelberg, Germany, 2002; pp. 156–165. [CrossRef]
21. Campagna, M. Sec 4: Elliptic Curve Qu-Vanstone Implicit Certificate Scheme (ECQV). Standards for Efficient Cryptography, Version. 2013; 1. Available online: www.secg.org/sec4-1.0.pdf (accessed on 24 November 2021).
22. Ha, D.A.; Nguyen, K.T.; Zao, J.K. Efficient Authentication of Resource-Constrained IoT Devices Based on ECQV Implicit Certificates and Datagram Transport Layer Security Protocol. In Proceedings of the Seventh Symposium on Information and Communication Technology, Ho Chi Minh City, Vietnam, 8–9 December 2016; SoICT '16. Association for Computing Machinery: New York, NY, USA, 2016; pp. 173–179. [CrossRef]
23. Khan, A.G.; Basharat, S.; Riaz, M.U. Analysis of Asymmetric Cryptography in Information Security Based on Computational Study to Ensure Confidentiality during Information Exchange. *Int. J. Sci. Eng. Res.* **2018**, *9*, 992–999.
24. Bokhari, M.U.; Shallal, Q.M. A Review on Symmetric Key Encryption Techniques in Cryptography. *Int. J. Comput. Appl.* **2016**, *147*, 43–48.
25. Li, H.; Dán, G.; Nahrstedt, K. Portunes+: Privacy-Preserving Fast Authentication for Dynamic Electric Vehicle Charging. *IEEE Trans. Smart Grid* **2017**, *8*, 2305–2313. [CrossRef]
26. Huang, X.; Xu, C.; Wang, P.; Liu, H. LNSC: A Security Model for Electric Vehicle and Charging Pile Management Based on Blockchain Ecosystem. *IEEE Access* **2018**, *6*, 13565–13574. [CrossRef]
27. Kim, M.; Park, K.; Yu, S.; Lee, J.; Park, Y.; Lee, S.-W.; Chung, B. A Secure Charging System for Electric Vehicles Based on Blockchain. *Sensors* **2019**, *19*, 3028. [CrossRef]
28. ElGhanam, E.; Ahmed, I.; Hassan, M.; Osman, A. Authentication and Billing for Dynamic Wireless EV Charging in an Internet of Electric Vehicles. *Future Internet* **2021**, *13*, 257. [CrossRef]
29. Babu, P.R.; Amin, R.; Reddy, A.G.; Das, A.K.; Susilo, W.; Park, Y. Robust Authentication Protocol for Dynamic Charging System of Electric Vehicles. *IEEE Trans. Veh. Technol.* **2021**, *70*, 11338–11351. [CrossRef]
30. Rabieh, K.; Wei, M. Efficient and Privacy-Aware Authentication Scheme for EVs Pre-Paid Wireless Charging Services. In Proceedings of the 2017 IEEE International Conference on Communications (ICC), Paris, France, 21–25 May 2017; pp. 1–6. [CrossRef]
31. Gunukula, S.; Sherif, A.B.T.; Pazos-Revilla, M.; Ausby, B.; Mahmoud, M.; Shen, X.S. Efficient Scheme for Secure and Privacy-Preserving Electric Vehicle Dynamic Charging System. In Proceedings of the 2017 IEEE International Conference on Communications (ICC), Paris, France, 21–25 May 2017; pp. 1–6. [CrossRef]
32. Roman, L.F.A.; Gondim, P.R.L. Authentication Protocol in CTNs for a CWD-WPT Charging System in a Cloud Environment. *Ad Hoc Netw.* **2020**, *97*, 102004. [CrossRef]
33. Fuchsbauer, G.; Vergnaud, D. Fair Blind Signatures without Random Oracles. In *Progress in Cryptology—AFRICACRYPT 2010*; Bernstein, D.J., Lange, T., Eds.; Lecture Notes in Computer Science; Springer: Berlin/Heidelberg, Germany, 2010; pp. 16–33. [CrossRef]
34. Li, F.; Xin, X.; Hu, Y. Efficient Certificate-Based Signcryption Scheme from Bilinear Pairings. *Int. J. Comput. Appl.* **2008**, *30*, 129–133. [CrossRef]
35. Xia, Z.; Fang, Z.; Gu, K.; Wang, J.; Tan, J.; Wang, G. Effective Charging Identity Authentication Scheme Based on Fog Computing in V2G Networks. *J. Inf. Secur. Appl.* **2021**, *58*, 102649. [CrossRef]
36. Roman, L.F.A.; Gondim, P.R.L.; Lloret, J. Pairing-Based Authentication Protocol for V2G Networks in Smart Grid. *Ad Hoc Netw.* **2019**, *90*, 101745. [CrossRef]
37. Kumar, G.; Saha, R.; Rai, M.K.; Buchanan, W.J.; Thomas, R.; Geetha, G.; Hoon-Kim, T.; Rodrigues, J.J.P.C. A Privacy-Preserving Secure Framework for Electric Vehicles in IoT Using Matching Market and Signcryption. *IEEE Trans. Veh. Technol.* **2020**, *69*, 7707–7722. [CrossRef]
38. Vaidya, B.; Mouftah, H.T. Multimodal and Multi-Pass Authentication Mechanisms for Electric Vehicle Charging Networks. In Proceedings of the International Wireless Communications and Mobile Computing (IWCMC), Limassol, Cyprus, 15–19 June 2020; pp. 371–376. [CrossRef]
39. Al-Shareeda, M.A.; Anbar, M.; Hasbullah, I.H.; Manickam, S. Survey of Authentication and Privacy Schemes in Vehicular Ad Hoc Networks. *IEEE Sens. J.* **2021**, *21*, 2422–2433. [CrossRef]
40. Braeken, A.; Touhafi, A. AAA—Autonomous Anonymous User Authentication and Its Application in V2G. *Concurr. Comput. Pract. Exp.* **2018**, *30*, e4303. [CrossRef]
41. Lu, Z.; Qu, G.; Liu, Z. A Survey on Recent Advances in Vehicular Network Security, Trust, and Privacy. *IEEE Trans. Intell. Transp. Syst.* **2019**, *20*, 760–776. [CrossRef]
42. Baee, M.A.R.; Simpson, L.; Foo, E.; Pieprzyk, J. Broadcast Authentication in Latency-Critical Applications: On the Efficiency of IEEE 1609.2. *IEEE Trans. Veh. Technol.* **2019**, *68*, 11577–11587. [CrossRef]
43. Almuhaideb, A.M. Re-AuTh: Lightweight Re-Authentication with Practical Key Management for Wireless Body Area Networks. *Arab. J. Sci. Eng.* **2021**, *46*, 8189–8202. [CrossRef]

44. Almuhaideb, A.M.; Algothami, S.S. Efficient Privacy-Preserving and Secure Authentication for Electric-Vehicle-to-Electric-Vehicle-Charging System Based on ECQV. *J. Sens. Actuator Netw.* **2022**, *11*, 28. [CrossRef]
45. Burrows, M.; Abadi, M.; Needham, R.M. A Logic of Authentication. *Proc. R. Soc. Lond. Math. Phys. Sci.* **1989**, *426*, 233–271. [CrossRef]
46. Park, K.; Park, Y.; Park, Y.; Das, A.K. 2PAKEP: Provably Secure and Efficient Two-Party Authenticated Key Exchange Protocol for Mobile Environment. *IEEE Access* **2018**, *6*, 30225–30241. [CrossRef]
47. Yu, S.; Lee, J.; Lee, K.; Park, K.; Park, Y. Secure Authentication Protocol for Wireless Sensor Networks in Vehicular Communications. *Sensors* **2018**, *18*, 3191. [CrossRef]
48. Park, K.; Park, Y.; Park, Y.; Goutham Reddy, A.; Das, A.K. Provably Secure and Efficient Authentication Protocol for Roaming Service in Global Mobility Networks. *IEEE Access* **2017**, *5*, 25110–25125. [CrossRef]
49. Odelu, V.; Das, A.K.; Choo, K.-K.R.; Kumar, N.; Park, Y. Efficient and Secure Time-Key Based Single Sign-On Authentication for Mobile Devices. *IEEE Access* **2017**, *5*, 27707–27721. [CrossRef]
50. Armando, A.; Basin, D.; Cuellar, J.; Rusinowitch, M.; Viganò, L. AVISPA: Automated Validation of Internet Security Protocols and Applications. Available online: https://www.ercim.eu/publication/Ercim_News/enw64/armando.html (accessed on 11 April 2022).
51. SPAN—Security Protocol Animator for AVISPA. Available online: http://people.irisa.fr/Thomas.Genet/span/ (accessed on 11 April 2022).
52. Von Oheimb, D. The High-Level Protocol Specification Language HLPSL Developed in the EU Project AVISPA. Proceedings of APPSEM 2005 Workshop, Frauenchiemsee, Germany, 12–15 September 2005; pp. 1–17.
53. Turuani, M. The CL-Atse Protocol Analyser. In *Term Rewriting and Applications*; Pfenning, F., Ed.; Springer: Berlin/Heidelberg, Germany, 2006; pp. 277–286. [CrossRef]
54. Basin, D.; Mödersheim, S.; Viganò, L. OFMC: A Symbolic Model Checker for Security Protocols. *Int. J. Inf. Secur.* **2005**, *4*, 181–208. [CrossRef]
55. Juels, A.; Brainard, J. Client Puzzles: A Cryptographic Countermeasure Against Connection Depletion Attacks. In Proceedings of the Networks and Distributed System Security Symposium (NDSS), San Diego, CA, USA, 1 January 1999.
56. Kilinc, H.H.; Yanik, T. A Survey of SIP Authentication and Key Agreement Schemes. *IEEE Commun. Surv. Tutor.* **2014**, *16*, 1005–1023. [CrossRef]