Vulnerability Analysis and Performance Enhancement of Authentication Protocol in Dynamic Wireless Power Transfer Systems

Tommaso Bianchi\textsuperscript{2}, Surudhi Asokraj\textsuperscript{1}, Alessandro Brighente\textsuperscript{2}, Mauro Conti\textsuperscript{2}, and Radha Poovendran\textsuperscript{1}

\textsuperscript{1} Network Security Lab, Department of Electrical and Computer Engineering, University of Washington, Seattle, WA 98195-2500, USA
\textsuperset{surudh22, rp3}@uw.edu
\textsuperset{1} Department of Mathematics, University of Padua, Via Trieste 63, Padua, 35121, Italy
tommaso.bianchi@studenti.unipd.it, \{alessandro.brighente, mauroconti\}@unipd.it

Abstract. Recent advancements in wireless charging technology, as well as the possibility of utilizing it in the Electric Vehicle (EV) domain for dynamic charging solutions, have fueled the demand for a secure and usable protocol in the Dynamic Wireless Power Transfer (DWPT) technology. The DWPT must operate in the presence of malicious adversaries that can undermine the charging process and harm the customer service quality, while preserving the privacy of the users. Recently, it was shown that the DWPT system is susceptible to adversarial attacks, including replay, denial-of-service and free-riding attacks, which can lead to the adversary blocking the authorized user from charging, enabling free charging for free riders and exploiting the location privacy of the customers. In this paper, we study the current State-Of-The-Art (SOTA) authentication protocols and make the following two contributions: a) we show that the SOTA is vulnerable to the tracking of the user activity and b) we propose an enhanced authentication protocol that eliminates the vulnerability while providing improved efficiency compared to the SOTA authentication protocols. By adopting authentication messages based only on exclusive OR operations, hashing, and hash chains, we optimize the protocol to achieve a complexity that varies linearly with the number of charging pads, providing improved scalability. Compared to SOTA, the proposed scheme has a performance gain in the computational cost of around 90\% on average for each pad.

Keywords: Electric Vehicles, Dynamic Wireless Charging, Authentication

1 Introduction

With more countries across the globe pushing for a transition from fossil-fuel combustion engines to electric automobiles in order to mitigate climate change,
there has been a rapid growth in the EV market over the past decade. The increasing popularity of EVs in the automobile industry is mainly because gas-powered vehicles are one of the largest sources of greenhouse gases. For example, in the United States, the transportation sector accounts for more than a quarter of total emissions [1]. Governments and automotive industries have been increasing their investments in technologies pertaining to EVs including but not limited to efficient charging infrastructures.

There exist two different types of charging infrastructures: static and dynamic. Although static charging systems are promising, they require the installation of isolated charging stations that must be interoperable with multiple automobile models of different manufacturers [2]. Compared to the traditional vehicles, the batteries used in static charging are heavy, expensive, and have a shorter lifespan. Static charging systems can also lead to range anxiety, which is the fear of running out of power in an EV and not being able to find a charging station in time to recharge the battery [3]. The above-listed reasons limit the widespread adoption of static systems, and hence of EVs. As a solution to these problems, dynamic wireless power transfer (DWPT) technology, that enables charging the EV while it is moving, is emerging to be a promising alternative [4]. Dynamic charging systems provide the EV the flexibility to charge when they are on the move with the help of charging pads (CPs) embedded under roads. To enable efficient utilization of this technology, we need to establish a Internet of Vehicles (IoV) where an EV can communicate and securely share information with the surrounding infrastructure and the CPs using Vehicle-to-Everything (V2X) communications. With the introduction of such dynamic charging infrastructures, secure and reliable mechanisms for authentication and billing of EVs by the DWPT systems are essential. The system must meet the security requirements of the customer utilizing the service such as location privacy, along with the physical and computational constraints presented by DWPT entities, for example, swift communication between the EV and the CP involved in the charging process.

Different approaches have been proposed in recent times to address the secure authentication of EVs. The works [5] and [6] proposed one of the first real-world scenario approaches in two steps, with additional billing schemes proposed in the second work. Different more complex approaches introduce the rise of security problems in the design implementation, if not carefully reviewed. An attacker could use the additional entities and message exchanges as entry points to build an exploit able to bypass the security and attack the system.

In this paper, we identify vulnerabilities in the existing protocol scheme [6] and we propose a novel, secure and efficient scheme with enhancements to the vulnerabilities and inefficiencies found in [6]. In addition, we provide a security analysis of the proposed scheme against common attacks in this environment.

Our contributions can be summarized as follows:

- we develop a new, efficient and secure protocol adopting exclusive OR operations, hashing and hash chains,
- we disclose vulnerabilities of an existing protocol,
we compare our solution to the SOTA solutions, demonstrating the improvement in the performance during authentication, using the Charm Crypto Python library to compute the primitive times [9].

The rest of the paper is structured as follows: in Section II, we describe the related work, considering proposals with the same system model, while in Section III, we present the model used in the proposed scheme. In Section IV, the authentication schemes [5] are presented, and the vulnerabilities are outlined in Section V. In Section VI, we describe our solution and improvements, providing a security analysis in Section VII and performance analysis in Section VIII. Section XI and X comprise discussion and conclusions, respectively.

2 Related Work

Over the past decade, several privacy-preserving authentication schemes for DWPT systems have been proposed. The authentication protocol needs to be communication and computationally lightweight, secure against different types of attacks and preserve the privacy of the user. The user identity is protected using pseudonyms that hide the real identity of the customer. From the simple physical representation of the system [7], researchers have developed a complete cryptography-based protocol managing the authentication of dynamic wireless charging for EVs.

One of the initial works proposed with this scenario is the protocol developed by Li et al. in 2015 [8]. Their scheme comprehends a DWPT system for authentication and billing making use of symmetric encryption with keys generated depending on the position in time and space of the EV. The protocol exchanges messages for authentication with each CP using a different key each time, resulting in a large overhead for the communication and computation of these keys.

Hussain et al. [6] published their scheme based on two different approaches described in detail later in the paper. They introduced the idea of CPs connected to the CSPA and the hash chain-based authentication and revocation of credentials to avoid the fraudulent use of the same. In this protocol, the EV exchanges multiple messages with the CP, which the CP relays to the CSPA and receives response from the CSPA. This leads to a large number of interactions between the CP and CSPA as well as increased utilization of the CP in the authentication scheme. As the CP is resource constrained and the contact time between EV and CP is limited, this authentication scheme might create a bottleneck for the protocol.

Two other works published by Zhao et al. [11] and Rabieh et al. [10] propose authentication schemes for a system model similar to the above works. The former uses public-key encryption with a signing and verification scheme provided by a Registration Authority (RA), and a bank in charge of the token provisioning for the charging requests. Each energy segment is assumed to transmit a constant amount of energy to the EV. The authors prove the scheme to be secure against free-riding, a particular attack in this scenario, in which the adversary tries to
pose as an authenticated user or taking advantage from the physic of the system for a free charge. Zhao et al. try to solve this problem checking the battery level of the vehicles at predefined time intervals. The latter scheme [10] is based on blind signatures, hash chains and XOR operations. In this case, additional to authentication and privacy, the authors address the double-spending attack, in which a malevolent user tries to abuse old credentials to get free charge.

Several other protocols employing multiple entities such as cloud servers (CS), fog servers (FS), pad owners (POs) and road-side units (RSU) have been proposed in [13,14,15,16]. However, all these approaches consider a decentralised infrastructure involving multiple mutual authentications between the EV and various other entities of the DWPT system. Such exchanges impose higher communication and computational cost on the EV. These systems have not addressed dynamic addition of RSUs and FS and the handover involved among them as well. Therefore, we consider a simple model that reduces attack surfaces as well as the number of mutual authentications while not compromising the critical security challenges and requirements required with the DWPT environment.

3 System and Adversary Model

We briefly describe the system model in Section 3.1 and the adversary model in Section 3.2.

3.1 System Model

We consider a system model for the dynamic charging infrastructure that comprises of the EV, CP, CSPA, and the RA in the authentication scheme. In the model, the CSPA is directly connected via a wired link to the RA and all the CPs. We provide the scheme in Figure 1. Authentication is achieved via direct communication between EV and CPs, where the CSPA can be involved or not, depending on the protocol scheme. The EV contains an On-Board Unit (OBU) that manages the cryptographic operations and securely stores the EV’s parameters. Such parameters include the EV’s real identity and the pseudonyms assigned to it. We consider the OBU secure and tamper-proof. The RA is responsible for publishing the parameters for the encryption scheme and generating the pseudonyms for the OBU at the time of registration. The communication between the different facilities and the OBU happens through Dedicated Short-Range Communication (DSRC) or wireless communication protocols.

3.2 Adversary Model

Our system considers both the EV and the CSPA malicious and not trustworthy. The adversary can compose, replay, intercept and forge messages, but they cannot decipher them without the correct cryptographic keys. The goal of the attacker is to infer the private key between the two entities, that allows them to get all the parameters of the charging process, and successively launch attacks.
in order to get a free charge or identify the vehicle, mining the privacy of the customer.

Only the RA is trustworthy and knows the true identity of the EVs, which is never revealed during the authentication process. Because of the symmetric key encryption, the wired communication network is considered secure; however, an adversary can connect to the network and sniff the traffic. If the protocol is poorly designed, an adversary may be able to infer useful information that could be used for further attacks, such as tracing or replay attacks. This adversary representation is formulated in the Dolev-Yao model [18].

Fig. 1. Network Model [5]

4 Reference Protocol

The authors in [5] proposed two different approaches for the model described in the previous section, namely Direct Mutual Authentication (DMA) and Pure Hash Chain-Based Authentication (PHA), reported in sections 4.1 and 4.2, respectively. Both the protocols rely on the same preliminaries and initialization phase. The first one is a direct authentication between EV and CPs, which can incur computation delays due to the memory constraints associated with the pads.

The second one is a faster hash-chain-based protocol that involves the participation of the CSPA, using the CPs as forwarders, thus incurring increased communication delay. The system uses El-Gamal encryption over an elliptic curve. The Department of Motor Vehicles (DMV) decides the parameters for the cryptographic scheme and generates its private and public keys. The DMV
is in charge of revocation, and the implementation of revocation is beyond the scope of this work but can be found well delineated in [5]. The cryptographic parameters are then stored in the Trusted Platform Module (TRM) of the EV at the time of registration, along with EV’s secret key, pseudonym generation key, and pseudonyms. The anonymous handles are generated using a secret counter and a secret incremental factor, unique for each EV. The generic formula is given by

\[
PS_{OBU}^i = \{(\alpha)_{K_{sym}} \parallel (\alpha \oplus ID)_{KV} \parallel n_i\}^{K_{DMV}}
\]

with \(\alpha = c_V + n_i \cdot inc_V\) as the update value and \(n_i\) as the counter of the current pseudonym. Furthermore, DMV generates an identifier for the vehicle as \(X_{OBU} = h(PS_{OBU}^1 \parallel PS_{OBU}^2 \parallel ... \parallel PS_{OBU}^n)\), stored in TRM along with the other parameters. The symbols used in the protocol are described in Table 1.

### 4.1 Direct Mutual Authentication

In the former protocol DMA, the OBU of the EV and CPs mutually authenticate each other without the involvement of the CSPA. The CSPA creates several master secret keys \(MSK_i\) and sends them to DMV. The DMV in turn sends \(X_{OBU}\) back to the CSPA. Before requesting to charge, the EV needs to register with the CSPA by sending \(PWD_{OBU}\) and \(X_{OBU}\).

\[
OBU \rightarrow CSPA : m_1 = \{PWD_{OBU}, X_{OBU}\}
\]

| SYMBOL     | DESCRIPTION                                      |
|------------|--------------------------------------------------|
| OBU        | On-Board Unit                                   |
| CSPA       | Charge Serving Provider Authority               |
| DMV        | Department of motor vehicle                     |
| CP         | Charging Pads                                   |
| ID         | Identity of the EV                              |
| ID_{CP}    | Identity of CP                                  |
| X_{OBU}    | Pseudonyms identifier for OBU                   |
| n_i        | Current count of EV pseudonym                   |
| inc_V      | Incrementing factor of EV                       |
| c_V        | Secret counter of EV                            |
| h(·), h_2(·)| collision-free hash functions                  |
| S_{OBU-CP} | Session key between CP and OBU                  |
| MSK_i      | Master Secret Key of CSPA                       |
| PS_{OBU}   | Pseudonym of OBU                                |
| r_{OBU,CP} | Random nonces                                   |
| KV         | Private key of EV                               |
| K_{sym}    | Pseudonym generation key of EV                  |
| K_{DMV}    | Private key of the DMV                          |
and the CSPA computes three values: $H_1$, $H_2$, $H_3$. The first is stored as a security parameter at CSPA while the rest are sent to the OBU along with the hash function $h()$ used to compute them:

- $H_1 = h(s \parallel X_{OBU})$,
- $H_2 = h^2(s \parallel X_{OBU})$,
- $H_3 = MSK_i \oplus H_1$,

with message,

$$\text{CSPA} \rightarrow \text{OBU} : m_2 = \{X_{OBU}, H_2, H_3, h()\}$$

On receiving this message, the EV can authenticate with the CPs without the help of the CSPA. The OBU begins the protocol by selecting a pseudonym $PS^i_{OBU}(i = 1, 2, ..., n)$ and computes the following parameters to forward them to the CP:

- $c_1 = h(H_2) \oplus PS^i_{OBU}$,
- $c_2 = h(PS^i_{OBU} \oplus X_{OBU})$, where $X_{OBU} = h(PS^1_{OBU} || PS^2_{OBU} || ... || PS^n_{OBU})$,
- $c_3 = h(h(PS^i_{OBU} \parallel c_2 \parallel H_3))$,
- $\delta_4 = r_{OBU} \oplus PS^i_{OBU}$,

with message,

$$\text{OBU} \rightarrow \text{CP} : m_3 = \{c_1, c_2, c_3, H_3, \delta_4\}$$

OBU sends all these parameters to CP along with $H_3$. CP then extracts the secret $H_1$ from $H_3$, computes $H_2$ in order to get $PS^i_{OBU}$ from $c_1$. The CP goes on to check the value of $c_2$ and $c_3$ for authentication. In the next step, CP selects a nonce $r_{CP}$, generates the session key $SK_{OBU-CP}$ following the generation of $ID_J = h(r_{OBU} \parallel ID_{CP})$ and sends back to the EV the following parameters:

- $c_4 = ID_J \oplus r_{CP}$,
- $c_5 = r_{CP} \oplus h(h(PS^i_{OBU}))$,
- $c_6 = h(r_{CP} \parallel c_4 \parallel c_5)$,
- $c_7 = H_1 \oplus h^2(PS^i_{OBU})$,

with message

$$\text{CP} \rightarrow \text{OBU} : m_4 = \{c_4, c_5, c_6, c_7\}$$

From these parameters, OBU can extract the $r_{CP}$ using $ID_J$, assuming that the OBU knows the value of $ID_{CP}$. In order to authenticate CP, OBU checks the values of $c_5$ and $c_6$ while $c_7$ is used to extract $H_1$ and stored as a security parameter. At this point, the two involved parties can establish a session key $SK_{OBU-CP}$ used later for the billing process. After the authentication is complete, the charging process begins.

According to the adversary model used, CSPA is not trustworthy and the use of $X_{OBU}$ is a threat for the privacy of the customer, in that is fixed for a large amount of protocol runs. In this way, a malicious CSPA could track the customer along the road and possibly identify the identity of the user.
4.2 Pure Hash Chain-Based Authentication

The second approach requires the involvement of CSPA and makes use of hash chains. This system lowers the computation cost required by OBU and CSPA but can incur a higher communication delay.

After the common initialization and pseudonym assignment, the DMV provides the OBU with the corresponding hash chain for each pseudonym:

\[ h(PS_{OBU}^i), h^2(PS_{OBU}^i), ..., h^n(PS_{OBU}^i) \]

The OBU registers with CSPA by sending the head of the hash chain of the current pseudonym. In order to authenticate with the CP, the OBU has to send the next hash chain value expected. The CP will forward the hash value to CSPA for authentication: the provider validates the received parameter against the value expected after using the hash operation as \( h(h^{n-1}(PS_{OBU}^i)) = h^n(PS_{OBU}^i) \).

Following this procedure, the CSPA replaces \( h^n(PS_{OBU}^i) \) with \( h(h^{n-1}(PS_{OBU}^i)) \). As described in the following section, this is a vulnerability that impacts the protocol’s security. This method expects the OBU to authenticate at each pad to receive the designated amount of energy. After authentication, the CSPA provides CP with a session key \( SK_{OBU-CP} \). The CP stores the session key in its database along with a timestamp.

5 Vulnerabilities and Attacks

As mentioned in the previous section, there is a possible attack vector in the Pure Hash Chain-Based approach. The generation of a hash chain is depicted in Fig. 2, while the intended use of a hash chain for authentication is depicted in Fig. 3. The problem is in how the CSPA updates the value for the next expected hash chain value:

\[ h^n(PS_{OBU}^i) \rightarrow h(h^{n-1}(PS_{OBU}^i)). \]

Instead of storing the current value received from the OBU \( h^{n-1}(PS_{OBU}^i) \), CSPA stores the hash value that is already in memory, resulting in the same hash chain value received at the start of the protocol being stored:

\[ h(h^{n-1}(PS_{OBU}^i)) = h^n(PS_{OBU}^i). \]

This behaviour can have two consequences:

\[ X \rightarrow h(x) \rightarrow h^2(x) \rightarrow h^3(x) \rightarrow \cdots \rightarrow h^n(x) \]

Fig. 2. Hash chain representation
The OBU currently participating in the protocol exchange is unable to authenticate itself further. This occurs because the hash value that the OBU sends, which could be say, $h^{n-2}(PS_{OBU})$, and the value that the CP or CSPA expects differ. Following the initial successful authentication, the CSPA repeatedly waits for the same value, resulting in an error.

The OBU could send the same $h^{n-1}(PS_{OBU})$ value for authentication indefinitely, and an attacker could eavesdrop on the packet and then pose as the authenticated vehicle, resulting in a successful free-riding through replay attack.

To mitigate this behaviour, it is sufficient to store the most recent value received from OBU, generalizing:

$$h^i(PS_{OBU}) \rightarrow h^{i-1}(PS_{OBU}).$$

Some parameters in the message exchanges during the DMA approach are ambiguous or can be eliminated because they add little to the protocol itself. The message $c_2$ sent by OBU to CSPA, which the authors claim is required for authentication and the subsequent verification of $c_3$ presents two issues. If CSPA wants to check the value $c_2$, it must know which $X_{OBU}$ to compare. CSPA is aware of all the registered values of $X_{OBU}$, but this may be a performance issue because it must brute-force the database to find the corresponding entry that matches the computation of $c_2$. The second inefficiency occurs when the CSPA extracts the value of $X_{OBU}$ and checks the validity of $c_2$ with this value. In this case, the computed and received values are identical because the extracted value is used for the exact computation. If this is the case, the extracted $X_{OBU}$ is never used in the protocol, rendering this step unnecessary. The billing section has an exception where the $X_{OBU}$ value can be substituted to avoid transmission.

Assuming OBU is aware of the value $ID_{CP}$, only the OBU can extract the value $r_{CP}$ from $c_4$ during the message exchange between CP and OBU because it is the only other entity aware of $r_{OBU}$ value. Furthermore, $c_7$ is used to extract and store the value of $H_1$, which is later on not used elsewhere. In this case, we know that $c_6$ and $c_7$ can be removed without affecting authentication.

Finally, as anticipated in the preceding section, the use of $X_{OBU}$ throughout multiple runs can expose the customer’s identity and allow tracking of it in different charging processes. This poses a serious threat to the DWPT system’s privacy requirements.
6 Revised Protocol

To address the vulnerabilities identified in the previous section, we modify the message exchanges between the OBU and the CSPA. To verify the veracity of the OBU registration later, we initially store a copy of the DMV’s database of vehicle pseudonyms at the CSPA without revealing the mapping between the OBU and its associated pseudonym. We modify the value of $X_{OBU}$ to $h_2(PS_{OBU}^i)$, where $h_2$ is a collision-free hash function provided by DMV. As a result, the value used to construct the authentication parameters is dynamic and cannot be used to track the user across different charging processes, thereby removing the issue highlighted in the previous section. This solution adds to the computational overhead at the start of each scheme run, but has no discernible effect on performance when compared to SOTA.

Following the first step, we modify the parameters sent during OBU-CP communication. To avoid authentication between the EV and the CP, the $m_2, m_3$ message exchange described in section 4 occurs between the OBU and the CSPA. As a result, we mutually authenticate with the system’s first level (the CSPA), and the OBU uses a hash chain to authenticate with the CPs. Consequently, $c_1$ retains its value while $c_2$ is completely removed. To authenticate with CP, we make the following changes:

- Instead of the original $c_3$, we use $c'_3 = h(h(PS_{OBU}^i) \parallel H_3)$;
- In $\delta_4$, we use the exclusive or $PS_{OBU}^i$ with $r_{OBU}$, a nonce generated for the current exchange and send $c'_4 = r_{OBU} \oplus PS_{OBU}^i$;
- OBU generates a nonce $N_{OBU}$ to use in order to generate the hash chain as $h^n(N_{OBU} \parallel PS_{OBU}^i)$.

CSPA can use the value of $H_1$ extracted from $H_3$ as in the original work, and consequently extract $PS_{OBU}^i$ from $c_1$, $r_{OBU}$ from $\delta_4$, and compare the results to $c'_3$. Following the Pure Hash Chain-Based Authentication approach, the hash chain will be used as authentication between OBU and each successive CP during the charging process. Summarizing the message exchange between OBU and CSPA again,

$$OBU \rightarrow CSPA : m'_4 = (c_1, c'_3, c'_4, c'_5, H_3).$$

where

$$c_1 = h(H_2) \oplus PS_{OBU}^i,$$
$$c'_3 = h(h(PS_{OBU}^i) \parallel H_3),$$
$$c'_4 = r_{OBU} \oplus PS_{OBU}^i,$$
$$c'_5 = h^n(N_{OBU} \parallel PS_{OBU}^i).$$

Similar to $r_{OBU}$, CSPA generates a nonce for the run of the protocol, $r_{CSPA}$. After removing $c_6, c_7$ and replacing $P, c_4$ and $c_5$, CSPA sends the following to OBU,
\[ CSPA \rightarrow OBU : m'_4 = (c'_6, c'_7). \]

where

\[ P' = h(r_{OBU} \parallel PS_{OBU}^{i}). \]
\[ c'_6 = P' \oplus r_{CSPA}, \]
\[ c'_7 = g((P' \oplus (r_{CSPA} - n))). \]

From this message, OBU can extract the value of \( r_{CSPA} \) and check the result of the exponentiation against \( c'_5 \). This approach necessitates the publication of the parameters \( g \) and \( n \) that are constant during the entire protocol. In the next phase, the CSPA sends the hash chain value provided by OBU to the CPs to authenticate and begin the charging process as in the hash chain-based authentication proposed in [5].

![Fig. 4. Diagram scheme of the first part of the improved protocol, the hash chain after this part is the same as in [5] except for the update function.](image)

7 Security Analysis

In this section, we provide an informal security analysis of our proposed scheme against the most common attacks for this scenario. In this environment, we consider the possibility of both the entities, the EV and the CSPA, being malicious, contemplating both passive and active attackers. The system provides mutual authentication between the OBU and CSPA, preserving the user’s privacy at the same time. The use of pseudonyms and the initial message exchange between the two entities meet these two requirements through the use of hash and exclusive or operations. The protocol is secure against the following attacks.
- Impersonation: it is difficult for an adversary to impersonate a different OBU in the authentication and charging process. The use of pseudonyms and their registration avoids the possibility of an attacker presenting as a different customer. During the authentication, sensitive information is hidden from an external party, making it impossible to infer the user parameters and nonces that is essential to pose as another user;

- Unlinkability: the generation of pseudonyms is restricted to the DMV. They are provided in a non-linear and complex manner, making it impossible for an external individual or CSPA to correlate a particular pseudonym to the corresponding user. Furthermore, linking different pseudonyms belonging to the same EV is unfeasible due to non-linearity and secret seeds in the pseudonym generation process;

- Man-In-The-Middle (MITM): In this scenario, an adversary cannot initiate an MITM attack due to the impossibility of impersonation and the presence of authentication messages containing secrets that are known and can only be used by the intended recipient. The attacker could only play the role of a forwarder unable to gain any advantage without modifying or forging messages posing as a different entity, as they cannot manage to extract the pseudonyms using the particular value $H_3$ and consequently $H_1$ and $H_2$ from the OBU authentication message;

- Free-riding: in this situation, a malicious user attempts to get a free charge without being authenticated. We are assuming that CPs are relatively shorter such that no two vehicles can be on the same pad at a particular time. The attack is mitigated as the hash chain used for authenticating with the CSPA can be generated only by the intended user. The adversary could listen to the current value to authenticate with the next pad. However, at the successive charging pad, CSPA waits for the next value of the hash chain to authenticate the OBU. This value cannot be computed by an unauthenticated, different vehicle;

- Double-spending: in this scenario, an EV that just utilized the service wants to use the same message again to get a free charge. The CSPA avoids this by storing previously used pseudonyms in the database for revocation in the event of a discrepancy.

8 Performance Analysis

In this section, we compare the computation and communication cost of different protocols with a similar system model against the revised protocol solution we proposed in section 6.

8.1 Methodology

In order to make the comparison, we count the number of operations in the authentication steps between OBU-CSPA and OBU-CP. As shown in Table 2, the total number of operations in the modified scheme is significantly reduced.
compared to the original protocol, even with the introduction of the exponentiation operation in both OBU and CSPA. We compute the reference time for two operations (hash and multiplication) using a simulation of the primitives in Python, using Charm-Crypto Library [9].

The exclusive or operation consumes less time, and hence, we exclude the time taken for exclusive or in the results. We simulate the experiment on a laptop with the following specifics: 8 GB of memory, 256 GB of SSD space, Intel Core i7-6500U @ 2.50 Ghz x 4, Ubuntu 20.04.3 LTS with 64-bit architecture. The simulation allows performing multiple executions to get a statistic of the time. We list the time required by each primitive in Table 2.

Table 2. Cryptographic primitive execution time

| Primitive | Average Time (ms) | Min. Time (ms) | Max. Time (ms) |
|-----------|-------------------|----------------|----------------|
| $T_{exp}$ | 0.110             | 0.102          | 0.630          |
| $T_{pair}$ | 0.884           | 2.763          | 0.833          |
| $T_h$    | 0.27              | 0.266          | 0.594          |
| $T_{ecm}$ | 1.352            | 1.352          | 1.352          |
| $T_{ver}$ | 1.449             | 1.449          | 1.449          |
| $T_{sig}$ | 0.992            | 0.992          | 0.992          |

8.2 Computation Cost

As listed in Table 3, the total time taken by our revised protocol is significantly lower than the original one even with the use of an exponentiation operation, primarily due to the reduction of the messages sent in the authentication exchange that presented redundancy. We also compare the revised protocol with other schemes with a similar infrastructure setup. In [10], a different approach comprising a digital signature is used, making the authentication part with the provider quite heavy. However, in the later part, a hash chain is used for authenticating with the CPs. In contrast to this, in Zhao et al. scheme, a digital signature is requested for each plate, making it computationally heavier than our proposed solution. The resulting optimization is around 5 times better against [10] and even more pronounced with [11].

8.3 Communication Cost

To compare the communication efficiency of the protocol, we compare our revised protocol with the original protocol. In Table 4, we report the cost that each step of authentication takes in terms of bytes. We consider that the pseudonym and
Table 3. Computational cost comparison

|                | Auth OBU                | Auth CP/CSPA/CMC | Hash-chain | Total time      |
|----------------|-------------------------|------------------|------------|-----------------|
| **Our Solution** | $5T_h + 3T_{XOR} + T_{exp}$ = 1.46 ms | $7T_h + 3T_{XOR} + T_{exp}$ = 2.0 ms | $n \times 0.27$ ms | 3.46 + $T_{Hash-chain}$ ms |
| Hussain [5]   | $6T_h + 6T_{XOR} = 1.62$ ms | $7T_h + 6T_{XOR} = 1.89$ ms | $n \times 0.27$ ms | 3.51$ \times n$ ms |
| Rabieh [10]   | $4T_{exp} + 4T_{ecm} + 2T_{ver} + T_{sig} + T_h = 10.01$ ms | $2T_{pair} + 4T_{ecm} + 4T_{exp} + T_{sig} + T_{ver}$ = 10.01 ms | $n \times 0.27$ ms | 20.02 + $T_{Hash-chain}$ ms |
| Zhao [11]     | $2T_{sig} + 2T_{ver} + T_h = 5.15$ ms | $T_{sig} + 2T_{ver} = 3.89$ ms | $n \times 0.27 + n \times (T_{sig} + T_{ver})$ ms | 0.04 + $T_{Hash-chain}$ ms |

the hash are 32 Bytes variables, and exponentiation is a 64 Bytes number with prime order $q$ of 512 bits.

Considering the integration of the hash chain at the CP level, we achieve better performance after the first constant part of the protocol. Following that, instead of the entire communication cost, we need only a message containing a hash, thus remarkably reducing the overall cost. Our protocol has minimal overhead stemming from the pre-authentication message exchange required to eliminate the usage of any parameter that can be used to track the user activity. Compared to the original scheme, which has a significant overhead at each CP, authentication between the EV and the CP in the proposed protocol consists of a hash chain value for each pad, which provides a significant advantage.

Table 4. Communication cost comparison

|                | **Our Solution** | Hussain et al. [5] |
|----------------|------------------|---------------------|
| Pre-authentication | $32 \times 3$ B | --                  |
| Auth OBU        | $32 \times 5$ B  | $32 \times 5$ B     |
| Auth CP / CSPA  | $32 + 64$ B      | $32 \times 4$ B     |
| Hash-chain      | $32$ B per CP    | --                  |
| Total cost      | $352 + 32 \times n$ B | $288 \times n$ B |

9 Conclusion

In this paper, we analyzed two authentication protocols for the dynamic charging of EVs and demonstrated vulnerabilities of the DWPT system in the proposed
scheme, which allows the adversary to attack the charging infrastructure or EV by eavesdropping, intercepting, and tampering with the exchanged messages. We propose an enhanced, lightweight and secure authentication protocol that improves system security by eliminating threats while lowering the computational costs and communication overhead of the system. The revised scheme protects the EV from adversarial attacks including but not limited to replay and denial-of-service attacks.

References

1. United States Environmental Protection Agency, Inventory of U.S. Greenhouse Gas Emissions and Sinks. https://www.epa.gov/ghgemissions/inventory-us-greenhouse-gas-emissions-and-sinks. Accessed 25 April 2022.
2. Pareek, S., Sujil, A., Ratra, S., Kumar, R.: Electric Vehicle Charging Station Challenges and Opportunities: A Future Perspective. In: 2020 International Conference on Emerging Trends in Communication, Control and Computing (ICONC3), 2020, pp. 1-6 (2020)
3. Mouftah, H.T., Erol-Kantarci, M., Sorour, S.: (Eds.). Connected and Autonomous Vehicles in Smart Cities (1st ed.). CRC Press.(2020)
4. Hutchinson, L., Waterson, B., Anvari, B., Naberezhnykh, D.: Potential of Wireless Power Transfer for Dynamic Charging of Electric Vehicles. In: IET Intelligent Transport Systems. pp. 3-12 (2019)
5. Hussain, R., Kim, D., Nogueira, M., Son, J., Tokuta, A., Oh, H.: A New Privacy-Aware Mutual Authentication Mechanism for Charging-on-the-Move in Online Electric Vehicles. In: 11th International Conference on Mobile Ad-hoc and Sensor Networks (MSN), 2015, pp. 108-115 (2015)
6. Hussain, R., Son, J., Kim, D., Nogueira, M., Oh, H., Tokuta A.O., Seo, J.: PBF: A New Privacy-Aware Billing Framework for Online Electric Vehicles with Bidirectional Auditability. In: Wireless Communications and Mobile Computing, vol. 2017, Article ID 5676030, 17 pages (2017)
7. Panchal, C., Stegen, S., Lu, J.: Review of static and dynamic wireless electric vehicle charging system. In: Engineering Science and Technology, an International Journal, Volume 21, Issue 5, 2018, Pages 922-937 (2018)
8. Li, H., Dán, G., Nahrstedt, K.: Portunes+: Privacy-Preserving Fast Authentication for Dynamic Electric Vehicle Charging. In: IEEE Transactions on Smart Grid, vol. 8, no. 5, pp. 2305-2313 (2017)
9. Charm-Crypto Python Library. https://jhuisi.github.io/charm/index.html. Accessed 20 April 2022
10. Rabieh, K., Wei, M.: Efficient and privacy-aware authentication scheme for EVs pre-paid wireless charging services. In: IEEE International Conference on Communications (ICC), 2017, pp. 1-6 (2017)
11. Zhao, X., Lin, J., Li, H.: Privacy-Preserving Billing Scheme against Free-Riders for Wireless Charging Electric Vehicles. In: Mobile Information Systems, vol. 2017, Article ID 1325698, 9 pages (2017)
12. Hamoudid, K., Adi, K.: Privacy-aware authentication scheme for electric vehicle in-motion wireless charging. In: 2020 International Symposium on Networks, Computers and Communications (ISNCC). IEEE, 2020, pp. 1–6 (2020)
13. 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. In: IEEE Transactions on Vehicular Technology, vol. 70, no. 11, pp. 11338-11351 (2021)
14. Roman, L. F., Gondim, P. R.: Authentication protocol in etns for a cwd-wpt charging system in a cloud environment. In: Ad Hoc Networks, vol. 97, p. 102004 (2020)
15. Gunukula, S., Sherif, A. B., Pazos-Revilla, M., Ausby, B., Mahmoud, M., Shen, X. S.: Efficient scheme for secure and privacy-preserving electric vehicle dynamic charging system. In: IEEE International Conference on Communications (ICC). IEEE, 2017, pp. 1–6 (2017)
16. Feng, X., Shi, Q., Xie, Q., Wang, L.: P2ba: A privacy-preserving protocol with batch authentication against semi-trusted rsus in vehicular ad hoc networks. In: IEEE Transactions on Information Forensics and Security, vol. 16, pp. 3888–3899 (2021)
17. Elghanam, E., Ahmed, I., Hassan, M., Osman, A.: Authentication and Billing for Dynamic Wireless EV Charging in an Internet of Electric Vehicles. In: Future Internet (2021)
18. Dolev, D., Yao, A.: On the security of public key protocols. In: IEEE Transactions on Information Theory, vol. 29, no. 2, pp. 198-208 (1983)