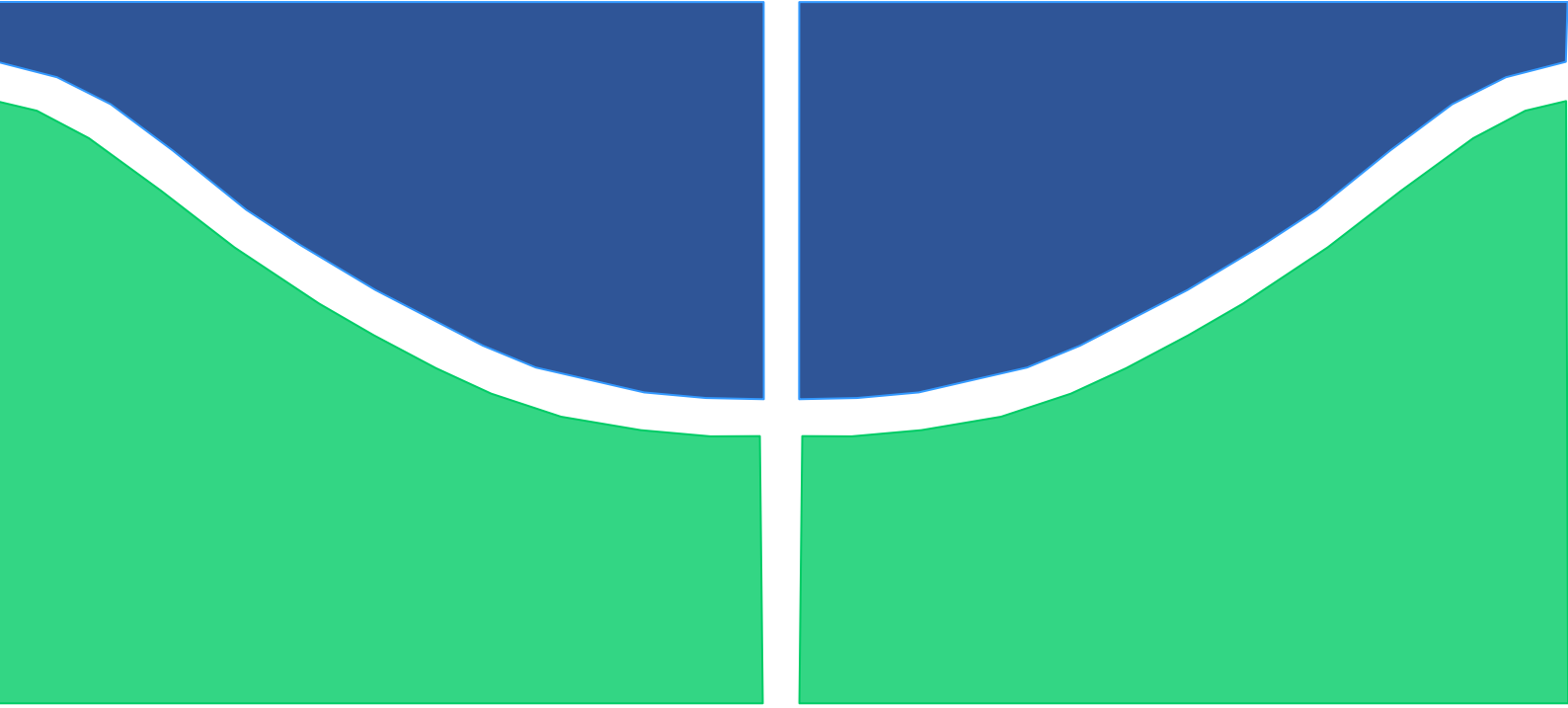


PROPOSAL OF AUTHENTICATION AND AUTHORIZATION PROTOCOLS FOR ELECTRIC VEHICLES CHARGING STATIONS

LUIS FERNANDO ARIAS ROMAN

DOCTORAL THESIS
IN ELECTRICAL ENGINEERING

DEPARTMENT OF ELECTRICAL ENGINEERING

FACULDADE DE TECNOLOGIA

UNIVERSIDADE DE BRASÍLIA

Proposal of Authentication and Authorization Protocols for Electric Vehicles Charging Stations

Luis Fernando Arias Roman

DOCTORAL THESIS SUBMITTED TO THE ELECTRICAL ENGINEERING DEPARTMENT AT THE UNIVERSITY OF BRASÍLIA AS A PARTIAL REQUIREMENT TO OBTAIN THE DEGREE OF PHD IN ELECTRICAL ENGINEERING.

APROVADA POR:

Paulo Roberto de Lira Gondim, D.C., FT/UnB _____
Orientador

Luiz Carlos Pessoa Albini, PhD, INF/UFPR _____
Examinador Externo

Renato Mariz de Moraes, PhD, CIN/UFPE _____
Examinador Externo

Marcelo Menezes de Carvalho, PhD, FT/UnB _____
Examinador Interno

Brasília/DF, Julho de 2023.

FICHA CATALOGRÁFICA

ROMAN L. F.

Proposal of Authentication and Authorization Protocols for Electric Vehicles Charging Stations [Distrito Federal] 2023.

xvii, 130p., 210 x 297 mm (ENE/FT/UnB, Doutor, Tese de Doutorado – Universidade de Brasília.

Faculdade de Tecnologia.

Departamento de Engenharia Elétrica

1. Authentication 2. Protocol

3. Cryptography 4.Security

I. ENE/FT/UnB II. Título (série)

REFERÊNCIA BIBLIOGRÁFICA

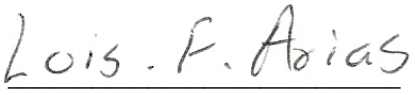
ROMAN L. F. (2023). Proposal of Authentication and Authorization Protocols for Electric Vehicles Charging Stations. Tese de Doutorado em Engenharia Elétrica, Publicação PPGEE 198/23 Departamento de Engenharia Elétrica, Universidade de Brasília, Brasília, DF, 130p.

CESSÃO DE DIREITOS

AUTOR: Luis Fernando Arias Roman.

TÍTULO: Proposal of Authentication and Authorization Protocols for Electric Vehicles Charging Stations.

GRAU: Doutor ANO: 2023



Luis Fernando Arias Roman
Qc7, Rua D, Casa 14.
71687420 Brasilia – DF – Brasil.

AGRADECIMENTOS

Quero agradecer primeiro à minha querida esposa Dianita Moon por seu apoio e incentivo para alcançar minhas metas diante de todas as dificuldades e momentos difíceis, sem ela não seria possível ter iniciado, nem terminado minha formação como doutor. Agradeço também à minha filha "Dandarita" quem chegou na minha vida no momento justo, dificultando o estudo, mas me enchendo de amor, e ao resto da família Luly, Campanita e Bento por todo o amor e companhia nas noites de estudo.

À minha mãe Reina, quem com muito esforço e amor cuidou de mim e dos meus irmãos, nos ensinando a ser boas pessoas e ter a coragem de enfrentar os problemas. Este doutorado é um logro também para ela.

Aos meus irmãos Jhon, Oscar e Veronica que me ofereceram todo o seu amor e apoio, e ao meu sobrinho Juan e à minha cunhada Daniela que me encheram de alegria.

Ao meu orientador Paulo Gondim, pela paciência, conselhos, o apoio e por acreditar no meu esforço, sem suas palavras e correções precisas, eu não teria conseguido chegar a esta tão esperada instância. Obrigado por sua orientação e todos os seus conselhos, vou carregá-los para sempre em minha memória no meu futuro.

Para meus sogros Alvaro e Martha pelo apoio e a boa energia que me ajudo a seguir na frente nos momentos difíceis.

Aos meus tios e primos que são minha fortaleza e o exemplo a seguir. Ao meu pai, pelos conselhos e apoio.

Aos meus amigos no Brasil Harry, Andres e Tatiana, pela torcida e amizade, são nossa família no Brasil.

Aos meus melhores Amigos Daniel, Carlos, Guevarita, Pipe e Carlos Mario pela companhia e suas boas energias, eles nunca estão longe.

Para meus colegas e amigos Albarci, Gabriel, Henrique, Julian, Junior e Marco pela torcida e apoio durante tempo de estudo e pesquisas.

Ao pessoal do Laboratório de Televisão Digital da UnB e ao Programa de Pós-graduação em Engenharia Elétrica da UnB.

À Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES), pela bolsa de estudos.

"*No vivas para ser por temor a presa de otros sueños*

Se vive uma vez para ser eternamente libre"

Kraken

Dedicado a

Minha esposa Diana Moon, a Minha filha Dandara e a Minha mãe Reina

RESUMO

PROPOSTA DE PROTOCOLOS DE AUTENTICAÇÃO E AUTORIZAÇÃO PARA POSTOS DE CARREGAMENTO DE VEÍCULOS ELÉTRICOS

Autor: Luis Fernando Arias Roman

Orientador: Paulo Roberto de Lira Gondim

Programa de Pós-graduação em Engenharia Elétrica

Brasília, mês de Julho (2023)

O carregamento sem fio para veículos elétricos (“Electrical Vehicles” - EV) enquanto o veículo está em movimento ganhou atenção especial como um novo serviço. Este serviço é suportado por tecnologias de transferência de energia sem fio (“Wireless Power Transfer” - WPT), que promovem o carregamento durante a condução (“Charging while Drive” - CWD) por meio de indução magnética de bobinas instaladas no solo. No entanto, o serviço também trouxe novos desafios, incluindo a segurança do sistema, que devem ser resolvidos.

O sistema de carga CWD-WPT deve garantir a privacidade, o anonimato, a integridade e a disponibilidade dos dados armazenados ou em trânsito pelo sistema, sendo necessária a implementação de um controle de acesso por meio da autenticação do usuário para garantir a segurança e privacidade dos dados. O processo de autenticação do usuário é fundamental para o sistema de carga CWD-WPT, e os protocolos utilizados para esta tarefa devem garantir o acesso de usuários válidos ao sistema e resistir a ataques de segurança.

Esta tese de doutorado aborda o projeto de protocolos de autenticação e autorização de uma estação de carregamento CWD-WDP baseada em nuvem, que garante a segurança da informação de forma mais eficiente, na maioria dos casos, em termos de custos de comunicação, computação e energia, em comparação a outros protocolos publicados. Esta tese apresenta 4 (quatro) protocolos para autenticação e controle de acesso de EVs em uma estação de carregamento CWD-WPT integrada em uma infraestrutura VANET (“Vehicular Ad Hoc Network”) baseada em nuvem.

O 1º, 2º e 4º protocolos foram projetados com base em uma estação de carregamento com controle centralizado, enquanto o 3º protocolo projetado com base em uma estação de carregamento com controle descentralizado. O 1º protocolo foi construído principalmente com o uso de criptografia baseada em emparelhamento bilinear e cadeia de “hash”. O 2º protocolo é uma variante do primeiro, cujo principal diferencial é a adoção de um novo esquema criptográfico baseado em mapas caóticos e árvore binária para controle de acesso no sistema. Seu desempenho em relação a métricas como custos computacionais, de comunicação e de energia é melhor do que outros esquemas e garante a autenticação mútua entre os EVs e todas as entidades do sistema.

Por outro lado, o 3º protocolo foi projetado em uma arquitetura de carregamento CWD-WPT descentralizada, e os esquemas criptográficos utilizados são mapas caóticos e cadeia de “hash”. Foi empregado “blockchain” para a criação e gerenciamento de grupos e autenticação e controle de acesso do EV na estação de carregamento CWD-WPT. De acordo com os resultados, o protocolo baseado em “blockchain” obteve melhor desempenho computacional, energia e recursos de segurança em comparação com outros protocolos.

A arquitetura do sistema considerada para o desenho do 4º protocolo possui um esquema hierárquico segundo o qual o sistema de confiança, o blockchain e o sistema de cobrança CWD-WPT são gerenciados na nuvem tradicional, enquanto a computação em névoa gerencia as RSUs (“Road Side Units”) das estações de carregamento. O protocolo usa confiança computacional para validar a forma de autenticação no sistema. Se a confiança do usuário estiver acima de certo nível, o processo de autenticação no sistema torna-se mais leve e rápido, sem descuidar da segurança das comunicações. A utilização de mapas caóticos se mostrou vantajosa em termos de desempenho de execução e tem promovido uma rápida criação de chaves de sessão e assinaturas digitais com baixo custo computacional. Por outro lado, o blockchain fornece às redes VANET transparência em seu funcionamento, resistência a ataques e uma validação rápida e eficiente das credenciais do usuário no processo de autenticação para autorizar ou negar seu acesso ao sistema. Também garante uma alta disponibilidade do serviço devido ao seu design descentralizado.

A segurança dos protocolos propostos foi verificada analiticamente, sendo garantidas propriedades como autenticação mútua, acordo de chaves, confidencialidade, integridade, privacidade, sigilo direto perfeito e sigilo perfeito reverso; além disso, a análise destaca a resistência a ataques ao sistema, como injeção, repetição (“replay”) de mensagens, chave conhecida, negação de serviço (Denial of Service" - DoS, modelo OSI de 2-3 camadas), Homem no meio (“Man in the Middle” – MitM), mascaramento, personificação, desvinculabilidade (“unlinkability”), gastos duplos, resistência à adivinhação de senhas, vazamento de números aleatórios e informações privilegiadas. Finalmente, uma verificação formal de segurança foi realizada usando a ferramenta AVISPA.

ABSTRACT

PROPOSAL OF AUTHENTICATION AND AUTHORIZATION PROTOCOLS FOR ELECTRIC VEHICLES CHARGING STATIONS

Author: Luis Fernando Arias Roman

Supervisor: Paulo Roberto de Lira Gondim

Programa de Pós-graduação em Engenharia Elétrica

Brasília, month of July (2023)

Wireless charging for electric vehicles (EV), while the vehicle is in motion, has gained special attention as a new service for such vehicles. It is supported by wireless power transfer (WPT) technologies, which promote charging while driving (CWD) through magnetic induction from coils installed on the ground. However, the service has also led to new challenges, including system security, which must be met.

The CWD-WPT charging system must guarantee the privacy, anonymity, integrity, and availability of data stored or in transit through the system, thus requiring the implementation of an access control through user authentication towards ensuring data security and privacy.

The user authentication process is fundamental for the CWD-WPT charging system, and the protocols used for this task must guarantee the access of valid users to the system and resist security attacks.

This doctoral thesis addresses the design of authentication and authorization protocols of a cloud-based CWD-WDP charging station, which guarantees the security of information, in most cases, in a more efficient way in terms of costs in communication, computing and energy, compared to other published protocols.

This thesis presents 4 (four) protocols for the authentication and access control of EVs in a CWD-WPT charging station integrated in a cloud-based VANET (Vehicular Ad Hoc Network) infrastructure. The 1st, 2nd, and 4th ones were designed on the basis of a charging station with centralized control, whereas the 3rd is devoted to a decentralized control.

The 1st protocol was built primarily with the use of bilinear pairing based cryptography and hash chaining. The 2nd is a variant of the first, whose main difference is the adoption of a new cryptographic scheme based on chaotic maps and a binary tree for access control in the system. Their performance regarding metrics such as computational, communication, and energy costs is better than that of other schemes, and ensures mutual authentication among EVs and all entities in the system.

On the other hand, the 3rd protocol was designed on a decentralized CWD-WPT charging architecture, and the used cryptographic schemes are chaotic maps and hash chain. Blockchain was employed for the creation and management of groups and authentication and access control of the EV in the CWD-WPT charging station. According to the results, the blockchain-based protocol achieved better computational performance, energy, and security features compared to other protocols.

The system architecture considered for the design of the 4th protocol has a hierarchical scheme according to which the trust system, the blockchain, and the CWD-WPT billing system are managed in

the traditional cloud, while fog computing manages the RSUs (Road Side Units) of the charging stations. The protocol uses computational trust to validate the form of authentication in the system. If the user's confidence is above a certain level, the authentication process in the system becomes lighter and faster, without neglecting the security of communications.

The use of chaotic maps has been advantageous in terms of execution performance and has promoted a fast creation of session keys and digital signatures at low computational costs. On the other hand, blockchain provides VANET networks with transparency in their functioning, resistance to attacks, and a quick and efficient validation of the user's credentials in the authentication process for authorizing or denying their access to the system. It also guarantees a high availability of the service due to its decentralized design.

The security of the proposed protocols was analytically verified, and properties such as mutual authentication, key agreement, confidentiality, integrity, privacy, perfect forward secrecy and perfect backward secrecy have been guaranteed; additionally, the analysis highlights resistance to system attacks such as injection, replay, known key, Denial-of-Service (DoS, 2-3 layers OSI model), Man-in-the-Middle (MitM), masquerade, impersonation, unlinkability, double spending, resistance password-guessing, random number leakage, and privileged insider. Finally, a formal security check has been carried out using AVISPA tool.

Summary

List Of Figures

List Of Tables

LIST OF SYMBOLS NOMENCLATURE AND ABBREVIATIONS

Symbol	Description
ID	Identification
PID	Pseudo identity
$H(\ )$	Hash function
X, x, S	Private key
Y, Q, y	Public key
k	Session key
σ, η	Digital signature
(J, L)	Blind signature
ϕ	Pre-key of session
τ	Number of RSUs per fog server
ψ	Number of pads per RSU
α, v	Seed
t, ts	Timestamp
VK	Verification key
hash chain request	Hash chain request
*	Multiplication operator
$\hat{e}$	Bilinear Pairing
CCS	Company Charging Server
RSU	Roadside Unit
HMAC	Hash-based message authentication code
P	Point of the elliptical curve
m	Message
T	Chebyshev polynomial map function
β	Chebyshev polynomial map degree
K	Group key
$Cert$	Digital Certificate
(θ, t, o),	2^{nd} protocol Ticket
k_w	3^{rd} protocol Ticket
λ	Seed
$p, n, e, d, \mathsf{c}, \gamma, q, \sigma$	Prime numbers
Tx	Blockchain transaction
$\mathcal{L}$	decay factor
$\wp$	current time
$\Im$	graph referent to social relationships

ACRONYMS LIST

AVISPA	Automated Validation of Internet Security Protocols and Applications
BAN	Burrows Abadi Needham
BFT	Byzantine Fault Tolerance
CA	Certification Authority
CCC	Charging Control Center
CCS	Company Charging Server
CCTV	Closed-Circuit Television
CL-AtSe	Constraint Logic Based Attack Searcher
CMC	Charging Management Center
CP	Charging Pad
CPLs	Charging Plates
CPS	Cyber-Physical Systems
CSP	Charging Service Provider
CSPA	Charging Service Providing Authority
CWD	Charge While Driving
CWD-WPT	Charge While Driving – Wireless Power Transfer
DH	Diffie Hellman
DLP	Discrete Logarithm Problem
DMV	Department of Motor Vehicles
DoS	Denial of Service
DPoS	Delegated Proof of Stake
DSA	Digital Signature Algorithm
DSRC	Dedicated Short-Range Communication
ECC	Elliptic Curve Cryptography
ECDSA	Elliptic Curve Digital Signature Algorithm
EMF	Extreme Malicious Feedbacks
EP	Energy Provider
EVs	Electric Vehicles
FADEC	Fast Authentication for Dynamic EV Charging
FS	Fog Server
GN	General Nodes
GTV	Global True Value
HLPSL	High Level Protocol Specification Language
HMAC	Hash-Based Message Authentication Code
IoT	Internet of Things
ITS	Intelligent Transportation System
JFK	Just fast Keying
KDC	Key Distribution Center
MET	Mobile Energy Transmitter
MitM	Man in the Middle
OBU	On-Board Unit
OFMC	On-the-Fly Model Checker
OLEV	On-line Electric Vehicle
PBFT	Practical Byzantine Fault Tolerance

PFS	Perfect Forward Secrecy
PoA	Proof of Activity
PoB	Proof of Burn
PoC	Proof of Capacity
PoS	Proof of Stake
PoW	Proof of Work
PRNG	Pseudo Random Number Generator
RBFT	Redundant Byzantine Fault Tolerance
RSA	Rivest, Shamir and Adleman
RSUs	Roadside Units
SC	Smart Contracts
SHA	Secure Hash Algorithm
SoC	State of Charge
SS	Service Station
TA	Trusted Authority
TH	Threshold
TRM	Tamper Resistant Module
V2V	Vehicle to Vehicle
VANET	Vehicular Ad Hoc Network
WAVE	Wireless Access in Vehicle System
WPT	Wireless Power Transfer

1. INTRODUCTION

1.1. Contextualization

The evolution of cyber physical systems (CPS) represents a significant trend characterized by the use of heterogeneous data and knowledge integration. It has rocked the beginning of Industry 4.0 (the 4th industrial revolution) through the application of advanced information and communication technologies towards increasing autonomy and global productivity in industrial systems [1][2][3].

The functionalities and new industrial services of Industry 4.0 are underpinned by the rapid development of the Internet of Things (IoT). IoT is a complex and heterogeneous ecosystem that interconnects various objects on a large scale to deliver innovative services such as drone-based ones, healthcare, smart grid capabilities and electric vehicles [4][5].

Several areas of industry, including medicine, agriculture, education, and transportation, project the use of IoT for providing new services [6]. In the area of transport, the popularity of electric vehicles (EVs) has increased in recent years due to the scarcity of fossil fuels and environmental reasons. According to the Organization for Economic Cooperation and Development, the transportation sector consumes over 50% of the world's oil and is responsible for the emission of approximately 20% of carbon dioxide worldwide. Although the adoption of EVs can improve the environment and reduce the oil dependency, several technological and operational challenges must be overcome[7][8].

Some of such challenges are battery life and the long duration of charging, which generate time and mobility restrictions for users [9][7][8]. Researchers have been working on the development of a new method of charge while driving (CWD) based on wireless power transfer (WPT) technology [10][11]. The operation of charging while drive through wireless power transfer (CWD-WPT) is dependent on the energy induced by a set of pads (coils with < 80% energy transfer efficiency [12]) in the battery of the EV. Because each pad, embedded in the road pavement, can induce only a small amount of energy in function of the speed of the vehicle, each CWD-WPT charging station must provide a large number of pads for the EV battery charge [10][13][8][14].

On the other hand, mobility is perhaps the most important feature of the CWD-WPT charging system, since several elements, such as location, vehicle type, access control, connection type, connection time, state of charge (SoC), privacy, and security [15] must be considered for the supply of the charging service.

For the treatment of mobility and connectivity among vehicles, a vehicular ad hoc network (VANET) is one of the networks that can be considered to support a CWD-WPT system [10],[11], [12]. VANETs have drawn the attention of researchers due to their large range of applications and services and a safe, efficient, trouble free and entertaining intelligent transportation system (ITS). They provide vehicles with an onboard communication unit called On Board Unit (OBU), through which they communicate with both other vehicles and the infrastructure via Roadside Units (RSUs). IEEE 802.11p standard provides the Wireless Access in Vehicle System (WAVE) protocol and the basic radio standard for dedicated short range communications (DSRC) at a 5.9GHz frequency [9][16].

Due to the technological evolution and exponential growth in the number of intelligent vehicles, traditional VANETs have faced flexibility, scalability, and other types of problems The integration of the cloud with VANET networks aims to solve problems of flexibility and scalability, as well as to foster the evolution and creation of new services. Cloud-based VANET communications are comprised of a number of elements and environments that integrate seamlessly to provide users with efficient, scalable, and secure services. To achieve this harmonic integration in cloud-based VANET networks, several authors have proposed layered systems with different focuses, where security is a layer that interacts throughout the system [17] [18] [19].

Cloud computing is a new paradigm that proposes allocating servers geographically, but next to the devices to collect, process, organize and store data in real time. Its use in vehicular networks tends to facilitate or provide a great variety of services, besides being a solution to reduce the costs of communication [19]. However, it faces several security challenges, which include data storage, computing, virtualization and network security issues, as well as access control, software security and trust management issues [20].

More specifically, cloud-based vehicular networks security is a challenging problem because of its additional characteristics of heterogeneity and the high volume of vehicles. According to Ziquia et al. [17] the most important security requirements for these networks are: authentication, data integrity, confidentiality, access control, non-repudiation, and availability.

The next generation VANETs must also support high mobility, low latency, real time services and connectivity, which cannot be provided by conventional cloud computing. An effective solution to vehicular network problems is the fusion of fog computing with cloud computing [16][15] [21][22], for extending to the edge of wireless networks the conventional paradigm of cloud computing and meeting requirements related to low latency, seamless mobility, data storage close to users and adequate localization of mobile devices. Moreover, the use of fog servers promotes a better mobility management of vehicles and redirectioning of mobile applications to the closest fog server [15].

Such a cloud environment creates a scalable and hierarchical architecture, which is convenient for the sake of distributed processing and storage capabilities. Therefore, two CWD-WPT charging station architectures were considered in the present study. The first has its operations center (Company Charging Server – CCS) in the cloud, which enables the control of several CWD-WPT stations, and the second has a local operations center (Charging Control Center – CCC).

In the first architecture, the CCS is installed in the cloud computing and connected to a group of secondary servers (fog servers – FS), where the fog computing is installed. Each FS groups several RSUs and each RSU groups several pads together. In the second architecture, the CCC is located on the roadside and directly controls the pads that induce energy to the EVs, and RSUs, FS, and a trust authority (TA) allocated in the cloud are the elements that enable the EV to securely communicate with the CCC to perform the charging.

The CWD-WPT charging technology in a cloud and fog computing environment can provide comfort and time optimization for EV users, if security, privacy, authentication and anonymity are considered. Mechanisms for EVs to enter a carrier charging service in a controlled and anonymous manner require efficient mutual authentication [21][22].

The main challenges for the design of an authentication protocol in a charging system are to minimize the protocol execution time and to ensure security and resistance to information attacks. A cryptosystem for the design or adoption of various cryptographic techniques (short signature, blind signature, key

exchange, and mutual authentication) must be chosen for the achievement of the desired efficiency, and is expected to be fast and ensure a good security level to the system.

Proposals for authentication protocols have been reported in the literature. Li et al. [14] and Hussain et al. [23] designed protocols which focus on mutual authentication between entity and preservation of privacy; however, the analysis of security problems is poorly detailed. Other proposals such as those presented by Gunukula et al. [24] and Rabieh and Wei [25] guarantee anonymous authentication, privacy, unlinkability and prevent double spending; however, they disregard some attacks that may affect the system. Other shortcomings the proposed protocols have in common is the lack of a formal verification and performance comparison with other schemes.

Among the several cryptographic systems used in recent years for providing security information is chaos-based cryptography, whose advantages include less computational complexity than the multiplications of the elliptical curve [26][27], protection to users' privacy, sensitivity to initial parameters, unpredictability, and boundness.

On the other hand, new technologies such as blockchain can solve problems of security attacks and information dissemination in VANET networks. Blockchain has emerged as a decentralized storage mechanism shared by multiple geographically dispersed nodes, but members of a same network. All nodes propagate and check the signed messages transmitted over the network and synchronize the data blocks chained with the use of hash headers created successively with the hash header of the previous block synchronized by a consensus mechanism. Due to such blockchain characteristics, systems can be autonomous, immutable and decentralized [28] [29] [30].

This doctoral thesis proposes four authentication protocols, designed with two different cryptographic systems for the administration and distribution of keys in a CWD-WPT charging system in a cloud and fog computing environment, which guarantee privacy and integrity of messages, mutual authentication between the EV and the CWD-WPT charging station and EV anonymity. The first protocol is based on bilinear pairing; the second, which is a variant of the first, is based on chaotic cryptography, the third is supported by blockchain and based on chaotic cryptography, and the fourth implements trust management for providing the system with lighter authentication to trusting EVs.

1.2. Motivation

Motivations for the development of this research involved, initially, the study of a new wireless dynamic charging service (or charging while driving CWD), through wireless power transfer (WPT) in electric vehicles. Such a type of wireless charging is important for extending the autonomy of EVs, offering comfort and reducing travel times for users.

The CWD-WPT recharging service is an important advance for vehicular networks, although several security challenges still must be solved. One of the most important challenges is defining an appropriate architecture and guaranteeing the privacy and anonymity of the system's users, this can be achieved by including the system in the cloud and by implementing authentication and access control protocols in the CWD-WPT charging stations.

On the other hand, the performance of the protocols used in the cloud-based CWD-WPT system is important for its functioning, specifically to guarantee the confidentiality and privacy of user data in a dynamic, heterogeneous context that requires quick responses.

The architectures considered for the CWD-WPT charging station and the encryption schemes are a fundamental part of access control, enabling the definition of the behavior of protocols such as messages exchanged between devices, number of bits per message, operations to be executed in each device, among others. The use of an encryption scheme or a mixture of several ones can lead to good results, thus, reducing communication, computational-and energy costs.

The development of authentication and authorization protocols based on new encryption schemes or amixture of several schemes has aimed at satisfying all security requirements of the cloud -based CWD-WPT system, reducing communication, computing, and energy costs and obtaining a high-performance secure service. Additionally, the protocols can be used for the authentication and control of other VANET services that must guarantee the efficiency, flexibility, confidentiality, and security of the system.

1.3. Objectives

1.3.1. General Objective

Proposal of authentication and authorization protocols in the VANET networks for CWD-WPT charging system, which preserve information security and perform well in comparison to other protocols already published.

1.3.2. Specific Objectives

- Identification and application of basic and advanced concepts of new techniques of protection, mainly related to confidentiality, privacy and integrity, and non-repudiation and availability in CWD-WPT system.
- Evaluation of different proposals of authentication protocols for further comparisons.
- Characterization and evaluation of different key data encryption and management schemes.
- Proposal of authentication protocols for CWD-WPT system, with a good performance;
- Validation of the proposed protocols by formal security verification techniques.

1.4. Contributions

Some contributions of this work can be highlighted:

- four authentication and authorization protocols, enabling privacy and integrity preservation as well as key agreement and distribution;
- design of two new CWD-WPT dynamic charging architectures based on a fusion of fog computing with cloud computing;
- preservation of the anonymity of EVs, since the protocols are based on download tickets purchased offline;
- use of cryptographic primitives, such as short signatures and blind signatures based on bilinear pairing and chaotic maps for authentication with no jeopardy to the true identity of the EV;
- use of blockchain for designing a new protocol for a CWD-WPT charging station;
- mutual authentication among the EV and all entities of the CWD-WPT charging station;
- a security analysis considering several attacks that can affect the system, with a larger number of attacks, when compared to other proposals;
- a performance comparison with other protocols, involving communication and computational costs;
- a formal security verification of the protocols by AVISPA tool.

In terms of publications directly related to this work, 2 (two) articles were published, one in the Ad Hoc Networks Journal (Appendix A), and another in an event: International Wireless Communications and Mobile Computing Conference (IWCMC) (Appendix B).

Two other articles have been submitted for evaluation and possible publication in well-reputed JCR-ranked journals, as appear in Appendix C (based on Chapter 5, mainly on section 5.2) and Appendix D (based on Chapter 6, mainly on section 6.2).

1.5. Thesis Statement

The main challenges for the design of an authentication protocol in a charging system are to minimize the protocol execution time and to ensure security and resistance to information attacks. A cryptosystem for the design or adoption of various cryptographic techniques (short signature, blind signature, key exchange, and mutual authentication) must be chosen for the achievement of the desired efficiency. Moreover, it is expected to be fast and ensure a good security level to the system.

In this thesis we focus on the proposal and evaluation of secure authentication protocols for access control for a CWD-WPT system in VANET networks, which allows minimizing computational, communication and energy costs, aiming to guarantee the operation and resource economy in the system service.

1.6. Methodology

The methodology used in the research considers the following phases:
Phase 1: bibliographic review on the CWD-WPT topic;
Phase 2: an in-depth study about CWD-WPT system security and proposed protocols;
Phase 3: an in-depth study of encryption, authentication and key agreement schemes;
Phase 4: proposal of protocols that meet necessary protection security;
Phase 5: calculation and comparison of computational and communications costs (the energy cost was also obtained and compared for one of the protocols);
Phase 6: formal validations of the proposed protocols.

The proposed protocols focus mainly on authentication, authorization and key agreement issues. Considering a general architecture of a CWD-WPT system and set of entities, served by a infrastructure of cloud-based communication. Some premises/assumptions related to possible insecure parts were adopted and, in function of possible threats and vulnerabilities, a set of security properties was considered objectives to be reached by the proposed protocols.

For the sake of comparisons among protocols, communication costs were evaluated, considering message flows and bandwidth consumption; additionally, computing costs were also evaluated, considering processing times of operations made by the protocols.

Finally, formal validation of the proposed protocols was accomplished, using a tool named AVISPA (Automated Validation of Internet Security Protocols and Applications) as well as some of the respective back ends and a graphical animator.

1.7. Organization

The remainder of the thesis is organized as follows:

Chapter 2 provides preliminary information for the understanding of the proposed protocols and the trust management scheme.

In Chapter 3 the state-of-the-art of wireless dynamic recharge systems is addressed, including their categorization and comparison, with a focus on topics related to information security, such as authentication of EVs in a Dynamic Charging System, VANET Security based on Blockchain and computational trust in VANETs, Smart Grid networks and CWD-WPT systems.

Chapter 4 describes the system and adversary models, proposes 2 (two) protocols for a centralized recharge system, and reports on a comparative cost-based evaluation and an analysis of security properties;

Chapter 5 contains the proposal of a third protocol, devoted to a decentralized recharge system, and reports on a comparative cost-based evaluation and an analysis of security properties.

Chapter 6 is devoted to a proposal of a trust management scheme or protocol, based on blockchain, and its evaluation, for a scenario of a centralized CWD-WPT system.

Finally, in Chapter 7 the conclusions of the work developed are presented, and future work is outlined.

2. BACKGROUND

This chapter is dedicated to the description of cryptographic techniques and other security tools that were used to create the protocols proposed in this work, in addition to performing a description of the properties and security attacks that can affect the system, an explanation of the methodology used to quantify the system performance, and finally a brief presentation of how the CWD-WPT charging system works.

2.1. Cryptographic Techniques

Assymmetric encryption: this technique that has two keys (public key and private key) allows the communication between two or more users to be encrypted with their public key (of public knowledge) and decrypted with the private key (only known by the owner of the public key). This technique allows the agreement of keys between two or more entities, and in this work it is used with cryptographic schemes of bilinear pairing (first protocol) and Chebyshev Chaotic Map (second protocol).

Symmetric encryption: this technique relies on a key (usually called a session key) that is used to encrypt and decrypt messages. In this work, we use asymmetric encryption to agree a symmetric session key between entities so that they can communicate securely and efficiently afterwards.

Digital signature: the digital signature (usually issued by a trusted entity) certifies that the owner of the message is a specific entity. In this work, this technique is used over the bilinear pairing (first protocol) and Chebyshev Chaotic Map (second protocol) encryption schemes.

Blind signature: the blind signature (usually issued by a trusted entity) refers that the message content was not known by the trusted entity, but the trusted entity certifies that the owner of the message is a specific entity. In this work, this technique is used over the bilinear pairing (first protocol) and Chebyshev Chaotic Map (second protocol) encryption schemes.

Hash Chain: this is a technique that allows the generation of encryption keys by successively applying the hash function over the hash of an initial value; it is used for rapid generation of keys for CWD-WPT charging station pads.

Blockchain: also called Distributed Ledger Technology (DLT), it is a decentralized and distributed technology defined as a group of blocks that contain the public record of all digital events occurred and communicated to collaborating entities.

The following subsections describe the fundamentals of cryptographic schemes and security tools considered for the generation of authentication and authorization protocols for CWD-WPT charging systems.

2.1.1. Bilinear Pairing

The birth of the bilinear pairing in the beginning was created as an attack method to cryptographic schemes based on elliptic curves. only until the year 2000 were published the first researches that used bilinear pairing as a solution to cryptographic problems and not as tool.

Bilinear pairing in cryptography has favored the creation of new and creative cryptographic protocols, such as: identity-based cryptography, short signatures, key agreement schemes, among others [31].

Bilinear pairing is defined as the projection of two points of additive set G_1 formed by points on an elliptic curve E of order $l \in Z_p^+$, towards a same point of a multiplicative set G_2 formed by the elements of order $l \in Z_p^+$. The discrete logarithm problem (DLP) is assumed hard in both G_1 and G_2. A mapping $\hat{e} = (G_1, +)^2 \rightarrow (G_2, \cdot)$ satisfies the following properties for all a, b $\in Z_q^*$ and $c, d \in G$ ([32]).

1) Bilinearity:

$$\hat{e}(a + c, d) = \hat{e}(c, d)\hat{e}(a, d), \quad (1)$$
$$\hat{e}(c, d + a) = \hat{e}(c, d)\hat{e}(c, a). \quad (2)$$

2) Non degeneration:

$$\hat{e}(c, d) \neq 1_{G_2}. \quad (3)$$

3) Computational Efficiency.

Bilinear pairings have other easily verifiable properties, such as:

1) $$\hat{e}(x, \infty) = 1 \text{ e } \hat{e}(\infty, x) = 1, \quad (4)$$

2) $$\hat{e}(c, -d) = \hat{e}(-d, c) = \hat{e}(d, c)^{-1}, \quad (5)$$

3) $$\hat{e}(ac, bd) = \hat{e}(d, c)^{ab}, \quad (6)$$

4) $$\hat{e}(c, d) = \hat{e}(d, c). \quad (7)$$

and can be used for data encryption, digital signatures and key agreements. In our protocol they are employed for the generation of digital signatures.

2.1.2. Digital Signatures

A digital signature is one of the most important cryptography-based resources. It indicates the owner or creator of a document or clarifies someone agrees on the content of a document. Some digital signatures are based on a public key that links the identity of the user with its public key, whereas others are based on the identity of the that generates the public key from the user's identity through a deterministic algorithm. The public key verification is based on the use of the user's identity, making this scheme more efficient. The first short bilinear pairing scheme was created by Boneh et al. [33], and from it were created a large number of signature schemes based on the coincidence for different applications[32]. Below is a description of the digital signature schemes used in our protocol.

2.1.3. Short Signatures

Short signatures work well in environments of memory and bandwidth restrictions. The most used signature schemes are RSA (Rivest, Shamir and Adleman) and DSA (Digital Signature Algorithm), however, the signatures they generate are long. For example, if the 1024 bit module is used, the signatures

of RSA and DSA are 1024 bits long. The bilinear pairing scheme provides short length signatures of approximately 160 bits with a security level similar to those of 1024 bit RSA and DSA signatures [32].

A signature scheme based on bilinear pairing commonly involves [32]:

– Initialization: Let $H: \{0,1\}^* \rightarrow G1$ be a map to point hash function. The secret key is $X, \in Z_q^*$, and the public key is $Y = X * P$ for a signer.

– Sign: Given secret key x and a message $m \in \{0,1\}^*$, compute signature

$$\sigma = X * H(m). \tag{8}$$

– Verify: Given public key $Y = X * P$, a message m and a signature σ, verify $e(P,\sigma) = e(Y, H(m))$.

2.1.4. Blind Signatures

Blind signatures have been widely used in digital payment schemes for the obtaining of the signature of a document without the signatory knowing the information of the document. Moreover, the user cannot obtain other valid signatures of the same document after an interaction with the subscriber. The scheme used for our protocol was created by Zhang et al. [34] and is called "ID Based Blind Signature and Ring Signature from Pairings". It is characterized by the use of an identity-based cryptosystem over bilinear pairings for the verification and authentication of the signed information without knowing the identity of the sender.

2.1.5. Hash Chain

Hash chain is a computational operation for the efficient authentication of one time passwords, extending the lifetime of digital certificates, building one time signatures, amongst other functions. It was used in this study for the authentication and creation of session keys [35].

A hash chain is generated by a hash algorithm, as SHA (Secure Hash Algorithm), through which a user randomly selects a seed (S) and calculates the entire key chain. Figure 1 shows the process of creation of keys with a chain hash.

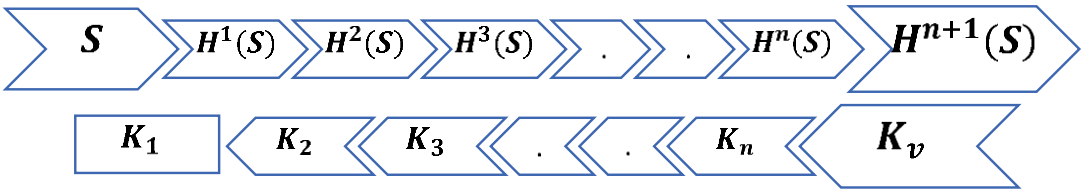


FIGURE 1. HASH CHAIN

The keys generated must be used in the opposite order of their generation, i.e., the last generated key K_n must be the first one used and the first key K_1 must be the last key used, such that an attacker listening to the channel cannot calculate a valid key from a used one. In our protocol, a public verification key K_v, is calculated applying $n + 1$ hashes to S for the validation of the keys. To verify a hash chain, an entity only applies successive hashes until it reaches the value of key K_v. If the key received after the application of n hash at maximum is not given the same value of the verification key, it is discarded.

2.1.6.Chebyshev Chaotic Map

Among the several cryptographic systems used in recent years for providing security information is chaos-based cryptography, whose advantages include less computational complexity than the multiplications of the elliptical curve [26][27], protection to users' privacy, sensitivity to initial parameters, unpredictability, and boundness. Chaotic sequences generated by the chaotic system commonly display non periodicity and pseudo randomness properties [36][37].

Chaotic map-based encryption has already been used in various scenarios for the design of authentication protocols, e.g., key agreement protocols ([18], [38]), user authentication protocols for multi server environments ([39], [40],[41]), group user authentication for social networks [42], authentication schemes in smart grid environments ([27], [43],[44]), session key agreement scheme in vehicular ad hoc networks [45], security in cloud environments ([46],[47]), among others. On the other hand, some authentication protocols for CWD-WPT systems proposed (e.g., [24] and [25]) exhibit security features such as privacy, integrity, anonymity and mutual authentication, but do not discuss some attacks that might disturb the system.

Chebyshev chaotic maps can be defined as:

Definition 1: assuming an integer value $\boldsymbol{n}$, a variable $\boldsymbol{x}$ in the [-1,1] interval, a Chebyshev polynomial $T_n(x)$: $[-1,1] \rightarrow [-1,1]$, of degree $\boldsymbol{n}$, is defined as:

$$T_n(x) = \begin{cases} cos(n * \arccos^{-1}(x)), x\epsilon[-1,1] \\ \cos(n\theta), \quad x = cos\theta \ ; \ \theta\epsilon[0,\pi]. \end{cases} \tag{9}$$

According to the previous definition, the recursive Chebyshev polynomials map $T_n: R \rightarrow R$ of degree $\boldsymbol{n}$, where $\boldsymbol{R}$ is the set of real numbers, and satisfies the following recurrence relationships:

$$T_0(x) = 1, \tag{10}$$
$$T_1(x) = x, \tag{11}$$
$$T_{n+1} = 2xT_n(x) - T_{n-1}(x), \text{ for } n \epsilon N. \tag{12}$$

Two important properties of Chebyshev's chaotic maps are shown below [42] [48]:

- Semi-group property: According to the validity of the semi-group property for Chebyshev polynomials in the $[-\infty, +\infty]$ interval (as shown in Zhang et al. [49]), it is also valid to consider

$$T_n(x) = \left(2xT_{n-1}(x) - T_{n-2}(x)\right) \bmod p, \tag{13}$$

where $\boldsymbol{n} \geq 2$, $\boldsymbol{p}$ is a large prime and $\boldsymbol{x} \epsilon (-\infty, +\infty)$.

$$T_r\left(T_s\left(x\right)\right) = T_{rs}\left(x\right) \tag{14}$$
$$= \cos(r * \arccos(\cos(s * asccos\ x))), \tag{15}$$
$$= \cos(r * s * \arccos(x)), \tag{16}$$
$$= \cos(s * r * \arccos(x)), \tag{17}$$
$$= \cos(s * \arccos(\cos(r * asccos\ x))), \tag{18}$$
$$= T_{sr}(x), \tag{19}$$
$$= T_s(T_r(x)), \tag{20}$$

where r and s are two positive integers, $s, r \epsilon Z^+$. Consequently, as in Zhang et al. [49],

$$T_r\left(T_s\left(x\right)\right) \equiv T_{rs}\left(x\right) \equiv T_s\left(T_r\left(x\right)\right) mod\ p. \tag{21}$$

- Chaotic property:

When $n > 1$, the Chebyshev polynomial mapping $T_n: [-1,1] \rightarrow [-1,1]$ of degree n is a chaotic map with invariant density:

$$f^*(x) = \frac{1}{\pi\sqrt{1-x^2}}. \tag{22}$$

Considering the Diffie-Hellman problems that are difficult to solve in polynomial time, the following definitions are met by Chebyshev polynomials ([26] [42]):

Definition 2. Chaotic maps discrete logarithm problem (DLP): Given two random numbers $\boldsymbol{x}$ and $\boldsymbol{y}$ belonging to the $[-\infty, +\infty]$ interval, the obtaining of a solution $\boldsymbol{w}$ that satisfies $y = T_w(x)$ is computationally infeasible.

Definition 3. Computational Chaotic Maps Diffie-Hellman Problem (DHP): Given $x, T_r(x)\ mod\ p$, and $T_s(x)\ mod\ p$, it is computationally infeasible to find $\boldsymbol{r}$ or $\boldsymbol{s}$ from $T_{rs}(x) mod\ p = T_{sr}(x) mod\ p$.

Figure 2 depicts a key agreement scheme between Alice and Bob, who aim at establishing communication through a secure channel. First, Alice and Bob agree on a seed $\boldsymbol{x}$ and a very large prime number $\boldsymbol{p}$ to start calculating their public keys. Alice chooses a number k_a (private key), applies Chebyshev's chaotic function to obtain a public key $A_k = T_{k_a}(x) mod\ p$, and sends it to Bob, who chooses a number $\boldsymbol{k_b}$ (private key) and applies Chebyshev's chaotic function to obtain a public key $B_k = T_{k_b}(x) mod\ p$. Alice and Bob exchange their public keys. Alice then applies Chebyshev's chaotic function to Bob's public key $\boldsymbol{k_a}$ times ($K_{sa} = T_{k_a}(B_k) mod\ p$) and Bob applies Chebyshev's chaotic function to Alice´s public key $\boldsymbol{k_b}$ times ($K_{sb} = T_{k_b}(A_k) mod\ p$). Upon finishing operations, both Alice and Bob obtain the same session key ($K_{sa} = K_{sb}$) to encrypt messages. Below is the mathematical proof.

$$K_{sa} = T_{k_a}(B_k) mod\ p, \tag{23}$$

$$K_{sa} = T_{k_a}\left(T_{k_b}(x) mod\ p\right) mod\ p, \tag{24}$$

$$K_{sa} = T_{k_a k_b}(x) mod\ p, \tag{25}$$

$$K_{sa} = T_{k_b k_a}(x) mod\ p, \tag{26}$$

$$K_{sa} = T_{k_b}(T_{k_a}(x) mod\ p) mod\ p, \tag{27}$$

$$K_{sa} = T_{k_b}(A_k) mod\ p, \tag{28}$$

$$K_{sa} = K_{sb}. \tag{29}$$

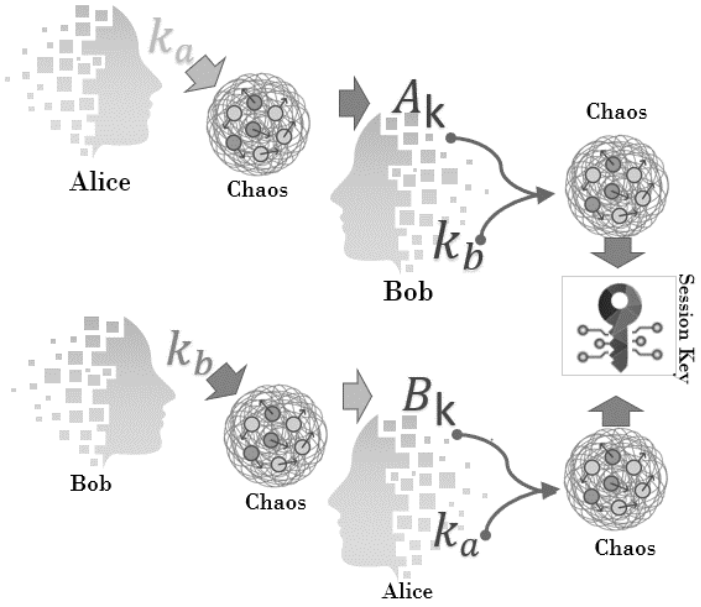


FIGURE 2. KEY AGREEMENT SCHEME BASED ON CHAOTIC MAPS

2.1.7.Chaos-based Signature

Towards signing a message in the protocol, we will use a signature scheme based on chaos and which has three phases, namely initialization, signing, and verification, described below:

- Initialization:
  Let $\boldsymbol{p}$ be a **large prime** number and hash function $H: \{0,1\}^* \rightarrow Z_q^*$ where $\boldsymbol{q} > \boldsymbol{p}$ and $gcd(p,q)$=1. The secret key is $\boldsymbol{s_{signer}} \in Z_p^*$ and the public key is $Y_{signer} = T_s(x) mod(p)$.

- Signing:
  For a message $m \in \{0,1\}^*$ and given secret key s_{signer}, compute value $h = H(m)$ and a signature

$$\eta = T_{s_{signer}*h}(x) mod(p). \tag{30}$$

- Verification:

  Given a public key Y_{signer}, a message m, and a signature η, verify:

$$T_{H(m)}(Y) mod(p) = T_{H(m)}\left(T_{s_{signer}}(x) mod(p)\right) mod(p), \tag{31}$$

$$= T_{H(m)*s_{siger}}(x) mod(p) \quad = \eta. \tag{32}$$

2.1.8.Chaos-based Blind Signatures

Below is the functioning of the chaos-based partially blind signature scheme [48], [50], according to three phases, described in what follows:

- Initialization:
  The initialization phase considers a large prime number *p*, the product $n = \overline{pq}$ of two primes $\bar{p}$ and $\bar{q}$ (taken as secret values of the system) and a factor of $p-1$; β, considered as an element in $GF(p)$ whose order module p is n; the multiplicative group G generated by β, and a hash function defined such that $H: \{0,1\}^* \rightarrow Z_p^*$.

  The signer randomly picks an integer $e \epsilon Z_n^*$ such that $gcd(e,n) = 1$, chooses a private key $x \epsilon Z_q^*$, an integer d satisfying $ed = 1(mod\ \varphi(n))$, and calculates a public key $z = T_x(\beta)$. The signer keeps the values confidential (d, x) and publishes values (n, z).

- Signing:
  We have assumed Alice (A) is the signer and Bob (B) is the user requesting the signature of document $\boldsymbol{m}$ from Alice**.**
  1. A chooses an integer $r < n$ such that $gcd(r,n) = 1$ and a value c and computes $\hat{t} = T_r(\beta) mod(p)$ such that $gcd(\hat{t}, n) = 1$. Signer A sends $\hat{t}$ and c to user B.
  2. When B receives A's message, he selects two blinding factors $(u, v) \epsilon Z_n^*$, and computes $t = T_{u+v}(\hat{t}) mod(p)$ such that $\gcd(t,n) = 1$; $\mu \equiv u^{-1} H(m)\, \hat{t}\, t^{-1}\ mod(n)$, and sends (μ, u) to signer A.
  3. A computes $\hat{k} \equiv (\mu\ x\ c\ r^{-1} + \hat{t}) mod(n)$ and sends it to B.
  4. B computes $k \equiv \hat{k}^{-e}(\hat{k}\ t\ \hat{t}^{-1}\ u + v\ t) mod(n)$ and sends it to A.
  5. A computes $\hat{R} \equiv (r\ k)^d\ mod(n)$ and sends it to B.
  6. B computes $R \equiv \hat{R}\ \hat{k}\ mod\ (n)$

  Finally, B obtains the signature (c, t, R) for message m.

- Verification:
  As demonstrated in [48], [50], verification of the signature of message m requires the equality of the following equation be satisfied:

$$\left[T_{R^e mod(n)}(\beta)\right]^2 + \left[T_{H(m)c\ mod(n)}(z)\right]^2 + [T_t(t)]^2 = \left(2T_{R^e}(\beta).T_{H(m)c}(z).T_t(t) + 1\right) mod(p). \quad (33)$$

With the use of the above described techniques and cryptographic schemes, the proposed authentication and authorization protocols manage to minimize computational costs and optimize the exchange of messages to perform the mutual authentication of entities and generate the session key, in addition to guaranteeing the confidentiality, privacy, integrity and availability of the CWD-WPT charging station service.

2.1.9. Blockchain

Blockchain creates a history of transactions by combining record blocks through encryption methods. Each collaborating entity can have a copy of the blockchain data thus, preventing the records contained in the blocks from being altered. Towards guaranteeing the anonymity of the entities that comprise the system, pseudo identities and public keys are used for interaction between entities. The first block generated in a blockchain is called genesis block; it contains initialization information commonly known by the other members of the system and serves as a basis for the generation of the next blocks in the

chain. Each block contains the cryptographic hashes of records, including information on the hash value of the previous block, thus, forming a data chain, i.e., a blockchain. [28][29]

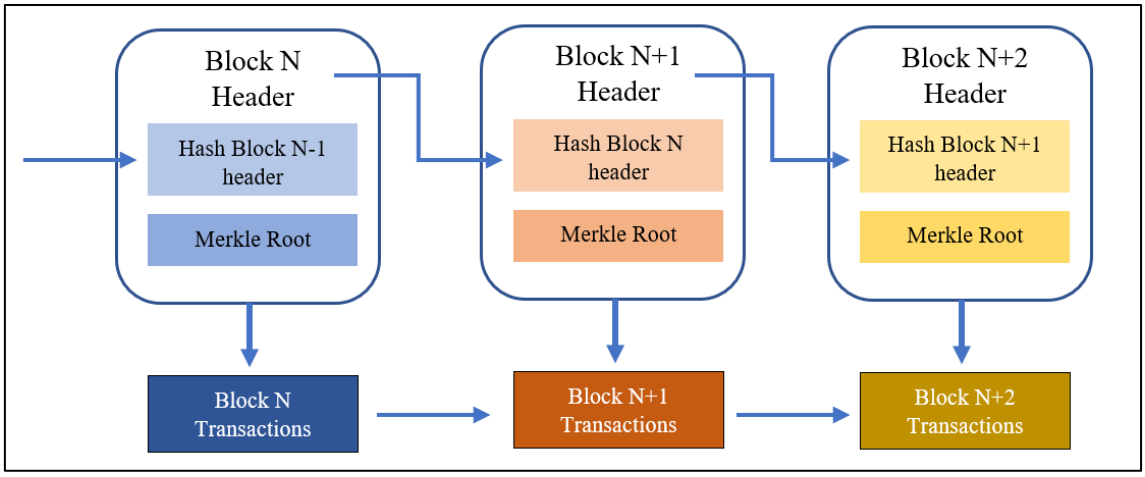


FIGURE 3. BLOCKCHAIN MODEL

All blocks are composed of a block header (the head of the block) and a block body. The former results from the execution of a hash function in the group of values that encompasses the header of the previous block, a random number (nonce), and Merkle root (binary hash trees).

On the other hand, the block body stores transaction details and other additional blockchain-related information (see Figure 3) [28].

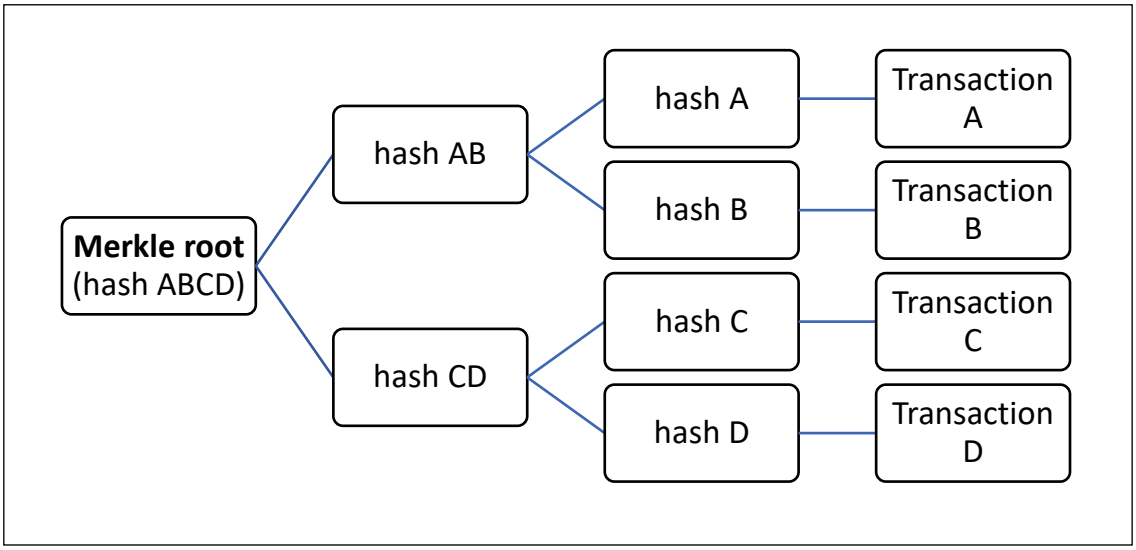


FIGURE 4. MERKLE TREE

The generation of Merkle root, one of the most important elements for the functioning of the blockchain, requires the application of an algorithm called Merkle trees, which groups all transactions to be registered in the block into pairs and applies a hash function for each transaction. The hash of each pair of transactions is concatenated for executing a hash function again. The result is concatenated with that of the hash function of the concatenation of another pair of transactions, and so on, until reaching the root of

the tree and a single hash value of the set of transactions to be recorded in the block. Figure 4 illustrates the process for the Merkle root calculation.

Three types of blockchain have commonly been considered:

- Public Blockchain: a non restrictive and permissionless blockchain, i.e., any entity (trusted or not) can access, validate transactions, and participate in consensus mechanisms, it is completely decentralized and used in systems such as Bitcoin, Ethereum, and Litecoin [29].
- Private Blockchain: because it is controlled by an organization or company, it is centralized, restrictive, and authoritative, and only entities predefined by the organization or company can maintain and validate the records. It is suitable for use in closed systems where all nodes (devices) trust each other.
- Consortium or hybrid blockchain: a decentralized blockchain comprised of several organizations or companies and used for semi closed systems composed of several companies such as a group of banks or government organizations.

A consensus mechanism, i.e., a set of rules that determine the contributions of blockchain devices (nodes), must be implemented for the acceptance of the new blockchain blocks by all members of the system. Below are the main consensus algorithms used in blockchain:

- Proof of Work (PoW): the system nodes compete to add a new block to the blockchain. The first node that finds a computationally heavy puzzle solution adds the new block and, consequently, receives a reward (in cryptocurrencies). Some applications that use it are Bitcoin, Litecoin, and Ethereum.
- Proof of Stake (PoS): all those who participated in the creation of a new block are rewarded (in cryptocurrencies) according to their contribution. Some applications that use it are PeerCoin, NXT, and Ethereum.
- Delegated Proof of Stake (DPoS): it is a version of PoS in which the participant with more money can delegate the signature of the blocks in the network, i.e., the participant of the largest balance can delegate the signature of the blocks and their profit to the members. It is used by BitShare as a consensus model.
- Proof of Capability (PoC): The PoC consensus system is similar to PoW. Miners compete to solve a difficult mathematical problem and thus generate a new block; however, PoC differs regarding the use of disk space to perform mining. Miners compute the blockchain once and store the results on disk.
- Activity Proof (PoA): miners create an empty block header and insert it into the network for other miners to perform a check. The nodes that receive the verified blocks insert them into the blockchain. The check is performed between miners and owners of the block.
- Practical Byzantine Fault Tolerance (PBFT): the main nodes that created and validated the new blocks are chosen. A consensus between them is reached through their exchange of messages. When a new block is generated by a node, this node sends a message to the master node, which sends it to the main nodes so that they check the validity of the block. Once it has been validated, the main nodes exchange messages informing on the acceptance or rejection of the new block. Each main node sends a message to the node that generated the block informing on whether the block has been accepted or rejected. It is used mostly by private blockchain systems.
- Redundant Byzantine Fault Tolerance (RBFT): RBFT is a variant of BFT. A new block is generated by a node, which sends a broadcast message to all main nodes for them to check its validity. Once the new block has been validated, the main nodes exchange messages

informing on the acceptance or rejection of the new block. Each main node sends a message to the node that generated the block informing on whether the block has been accepted or rejected.

The components for supporting the operation of the blockchain are [28]:

- Ledger: a record that stores blockchain history in a decentralized and unalterable way.
- Peer-to-Peer Network: a component that updates and stores the book. Each block in the network has a copy of the book. When the book is updated, all blocks of the network reach a consensus on the new book.
- Support services: in a public blockchain, all members have the same authority and any block can become part of the network. This service authenticates and authorizes identities of the blocks in the blockchain
- Smart Contract: an algorithm or module in which the rules and consequences of actions taken within the blockchain are defined and published, as in a traditional document, establishing obligations, benefits, and penalties due to the parties in different circumstances.
- Wallet: a component that stores credits and other user´s information.
- Events: a current state of the Peer-to-Peer Network and blockchain ledger, it notifies the user of the Smart Contract on the new addition of a new block to the blockchain and the accomplishment or removal of transactions.

2.2. Security properties and attacks

The properties and computer attacks considered to perform the security analysis of the proposed protocols are described below:

2.2.1. Security properties

- Integrity: guarantees the information is not modified during the journey from a sender to a recipient;
- Privacy and Anonymity: ensure users of a system control or influence information related to them that can be made available and stored and the way and to whom it can be disclosed;
- Confidentiality: ensures private and confidential information is not made available or disclosed to unauthorized individuals.
- Mutual authentication: ensures two parties validate their identities to each other before exchanging messages.
- Perfect Forward Secrecy (PFS): guarantees further valid keys of a system are generated and known only by the entities or valid users.
- Perfect Backward Secrecy (PBS): guarantees that no entity or user entering the system is able to decipher the previous messages exchanged in the system before they were joining;
- Unlinkability: guarantees only authorized entities identify the activities conducted by users in a system.
- Double spending: ensures some elements defined in a system are used only once and then discarded.

2.2.2. Informatics Attacks

- Privileged insider: consists in taking advantage of vulnerabilities in the internal controls and security policies of companies in charge of operating and managing the system;
- Random number leakage [51] : uses vulnerabilities in Pseudorandom number generator (PRNG) systems, which may have patterns in the random number generation sequence;
- Replay: occurs when an attacker intercepts communication packets between two valid entities and fraudulently retraces or forwards the message to the system. Consequently, information can be accessed through a simple forwarding of a captured message to the server;
- Man-in-the-middle: an attacker is positioned between two parties (sender and receiver) trying to communicate, intercepts the messages exchanged, and forwards his own messages to the parties, pretending to be a valid sender;
- DoS: aims to disable the use of the system by users, sending multiple false connection requests that saturate the system, and preventing valid requests from being answered. Although its avoidance in the 2 and 3 layers of the OSI model is highly complex - the attack can use all available bandwidth (to serve users) to cause service unavailability - it can be mitigated in the session layer if the bandwidth is not a limiting factor;
- Injection: consists of an attacker intercepting communication between two valid entities; Messages from a valid sender are intercepted and modified through the addition of information, and then forwarded to the recipient;
- Impersonation: an attacker tries to impersonate a valid entity and steal information from system users;
- Known key: attackers attempt to use old session keys already used to log into the system;
- Masquerade: an intruder tries to use a false identity to impersonate a legitimate system entity and then gain access to information.
- Resistance password-guessing: an attacker tries to find the access keys to the system through divination.

2.3. Cost Calculation

Since each protocol imposes costs related to its operation, a performance evaluation based on such costs can be made, allowing a comparison among different proposals of protocols. Such costs are commonly related to the usage of system resources, such as bandwidth, processing and energy; in a specialized view for the case of CWD-WPT charging systems, some variables must be considered, such as the number of EVs (denoted by the variable "$\boldsymbol{n}$"), the number of pads (represented by the variable "$\boldsymbol{\psi}$"), and the number of RSUs (represented by the variable "$\boldsymbol{\tau}$").

In what follows is the description of the methodology for the calculation of communications, computing, and energy costs of the protocols.

1. Communications Cost: refers to the number of bytes exchanged during communication between two entities. The byte values of each message element are summed and the value of each message is summed towards the total cost of a specific protocol or phase. The obtaining of the communication cost equation that defines the performance of the system requires the bytes of each message be calculated and then multiplied by the representative variable of the entity that generated the message.

2. Computational Cost: takes into account the time (in miliseconds or ms) required to perform unit operations, which are estimated according to the processing power of each entity. The

cost values are based on experiments conducted on common computational platforms and adopted for performance comparisons of authentication protocols. Both number and type of unit operations of each entity are identified and then multiplied by the representative variable of the entity that performed that operation. The execution times of some unitary operations were acquired from Tao et al. [52]. The obtaining of the execution times of the other unit operations required a search for jobs with the missing unit operation execution times and the development of a linear relationship between the execution time of a unit operation and the computing power of the system described in Tao et al. [52]. The execution costs of the Hash (T_{hash}) function for EV are based on Gunukula et al. [20], the execution costs for generating a signa- ture message (T_{g-sig}) and its verification (T_{v-sig}) are based on Rabieh et al. [21], and execution costs of the chaos (T_{chaos}) function is based on Cui et al. [45]. The execution costs of the hash function, signature message and message signature verification for RSU and FS were calculated analytically, taking 70% and 60%, respectively, from the cost of executing these operations to an EV.

3. Energy Cost: shows the importance of optimizing authentication protocols towards reducing the amount of energy used in systems. It is calculated from the computing costs that involve the time spent by a processor for calculating a variable, the power it used, and the energy in milijoules (Watts*miliseconds) spent by the system.

2.4. Wireless Power Transfer - WPT system

WPT refers to the charging of a device - in this case, an EV – with the use of resonant magnetic energy transfer. It makes the charging process more convenient, since no physical contact is established between the mains power supply and the electric battery of the EV due to the use of load coils on the ground (off board coil) and a receiver coil on the vehicle (on board coil). The vehicle is charged when its coil captures the energy contained in the electromagnetic fields generated by the charging coil. The process is illustrated in Figure 5.

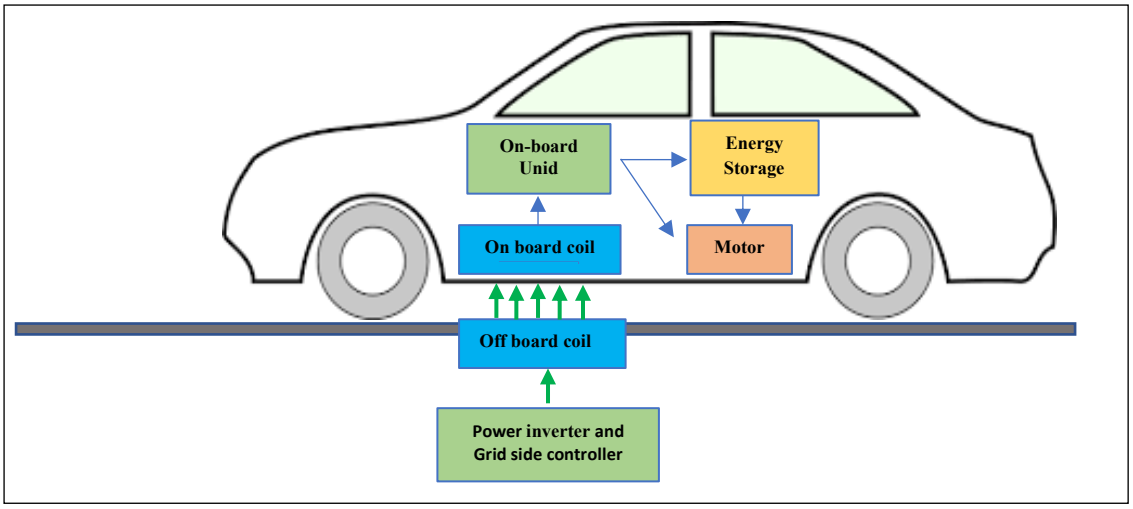


FIGURE 5: WIRELESS POWER TRANSFER SYSTEM - INDUCTIVE COUPLING

2.5. Computational Trust

Trust was introduced as a mechanism for detecting malicious attacks on systems. Trust has many context-dependent definitions, so the model of trust will depend on the system being designed, for example in the social sciences trust is the degree of subjective belief about the behavior of a particular entity. In other areas, trust is defined as the subjective probability that an agent will perform a given action, for VANETs, the definition of trust is similar to that of sociology, where trust is the degree of belief that an entity will perform tasks that it should, and has five basic properties: asymmetry, dynamicity, subjectivity, non-transitivity, and context dependence [53].

The difficulty of designing a trust model for a system consists in choosing (trust) factors that guarantee the objectivity of trust. Furthermore, trust models must consider accepting a degree of risk, as risk assessment is important for system trust.

One of the important factors to choose the trust model is the type of network topology, such as: Homogeneous networks (CCTV camera network), heterogeneous (IoT network), hierarchical networks (DNS server networks), static networks (sensor networks) , dynamic networks (VANET networks), among others. Each of these networks will have different challenges and contrasting methods for determining trust between entities.

On the other hand, in terms of security, conventional models (encryption, signatures, authentication, etc.) assume that attacks are carried out by an entity outside the network, but this approach is not enough to guarantee system security, as entities within the can be compromised and attacks can be performed from them[54].

A very important concept for trust models is "zero trust", this means that no entity is trusted until it has been verified. Entities' trust is dynamic and depends on time and on the events that take place.

The main contribution of trust models in system security is the dynamic access control to all internal / external entities of a system, depending on the scenario, the implemented model, the data provided for trust calculation. Trust models may also require authentication depending on the situation[54].

For the construction of trust models, two types of systems have been considered, centralized and decentralized. The centralized system has management advantages and reduced costs, but it has availability problem, because if the main system goes down, the system will fail. In the decentralized system availability problems are solved [55].

To provide a better understanding of what a trust model is and considering that trust models depend on several factors, below in figure 6 (based on the work of Mannix et al. [54]) a block diagram is shown that includes the most common and important features of trust models. We emphasize that the block diagram is not based on a specific trust model, but follows the concepts shown in the figure (fig. 6).

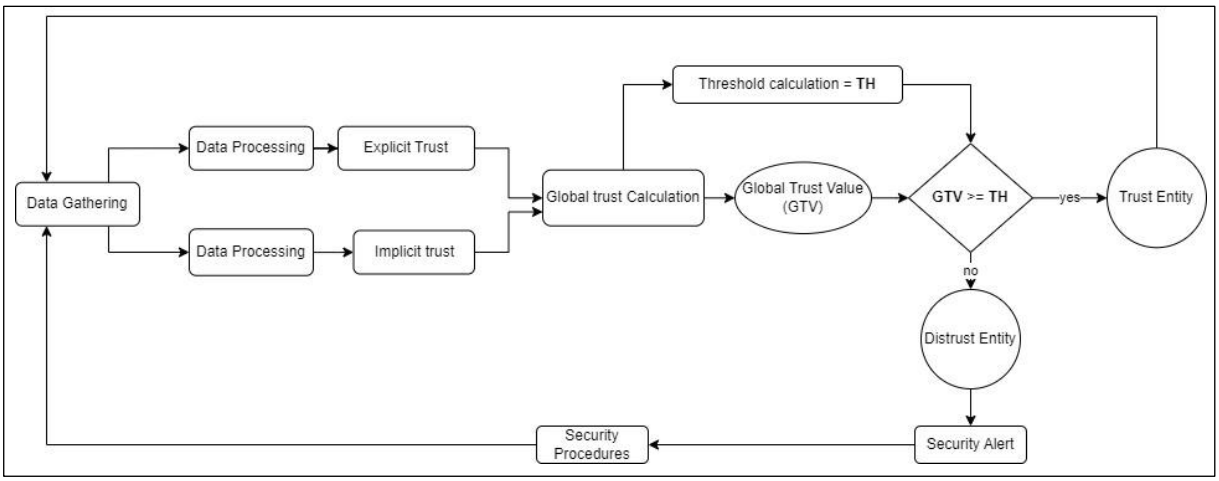


FIGURE 6: GENERAL TRUST MODEL

The initial phase of "Data Gathering" is one of the most important for the trust model. In this phase data is collected such as: entity's physical data (entity status, communication characteristics, hardware/software resources, among others); explicit trust data, referring to interactions between system entities, commonly evaluated with probabilistic methods; and implicit trust data, are explicit trust data obtained from another source, for example and A wants to communicate with B, A can ask C for the trust value he calculated to communicate with B.

In the "Data Processing phase", implicit trust and explicit data are pre-processed before performing the implicit and explicit trust calculation. This processing allows that, through mathematical models and depending on the system, some data are prioritized more than others for the calculation of the confidence value.

Then, the calculation of the "Global True Value" (GTV) is performed using a mathematical model to calculate the global confidence value considering the values found in the implicit and explicit trust. The result is a meaningful value used by the system to classify entities as malicious or not.

The "Threshold Calculation" in many models is a static value, but calculating a dynamic threshold value offers flexibility and adaptability to the system. After opting the threshold (Threshold - TH), the values and the GTV of the entity, these values are compared, and the system will decide if it is safe to interact with that entity or on the contrary it is a malicious entity.

If the entity is classified as "Trust Entity", the system will interact with it without restrictions until the time comes to perform a new validation. On the contrary, if it is classified as a "Distrustful Entity", the system will generate an "Security Alert" to be treated according to "Security Procedures", and then carry out a new validation of trust. Finally, the result will be stored historically for later confidence calculations.

The trust model is established from three aspects, namely explicit trust, implicit trust and global trust for assessing the reliability of the elements of the CWD-WPT system. Next, the process of calculating such variables is described, which is based on [56].

Explicit Trust: Explicit trust is a $TV_{i,j}^{h}$ nominal value assigned from the satisfaction that an entity $\boldsymbol{i}$ has when using a service presented by entity $\boldsymbol{j}$; for the present work, $\boldsymbol{i}$ will represent a EV that use the charging station, and $\boldsymbol{j}$ will represent the FS that manages the charging station. The $TV_{i,j}^{h}$ satisfaction value can be 0 or 1, depending on the absolute service dissatisfaction ($TV_{i,j}^{h} = 0$) or the complete

satisfaction ($TV_{i,j}^{h} = 1$) respectively. Considering the total number of interactions ($H_{i,j}$) between an EV (i) and entity j, the explicit trust of EV_i in entity j would be the weighted sum of historical classifications:

$$ETV_{i,j} = \frac{\sum_{h=1}^{H_{i,j}} TV_{i,j}^{h} . \Phi(t_h)}{\sum_{h=1}^{H_{i,j}} \Phi(t_h)}. \quad (34)$$

where $\Phi(th) = e^{-\mathcal{L}(\wp - t_h)}$ is the time function that captures the nature that more recent evaluations are more important than earlier ones, $\mathcal{L}$ is the decay factor, $\wp$ is the current time, and t_h is the *h-th* service time.

Implicit Trust: for EV_i this type of trust is calculated from trust assessments that other *EV* (i') have in relation to entity j. The ratings of all EVs are stored off-chain and can be denoted in a time string as:

$$RET_{i',j} = \langle \left(TV_{i',j}^{1}, t_1\right), \ldots, \left(TV_{i',j}^{h}, t_h\right), \ldots, \left(TV_{i',j}^{H_{i',j}}, t_{H_{i',j}}\right) \rangle. \quad (35)$$

EV_i can access personal recommendation data after the consent from the data owner and the blockchain network authorization. Malicious users commonly offer false and misleading recommendations and legitimate ones can give unfair and subjective feedback due to personal expectations and opinions.

For our case, the FS has authorization to access trust rating data between entity j and other EVs, but one has to consider that as soon as there are trusted users who offer fair ratings, there are also malicious users who can offer misleading or false ratings. To minimize the impact of false reviews or the credibility of all reviews should be verified, firstly detecting extreme malicious feedbacks (EMFs) and validating normal feedbacks. Specifically, rating beyond a $\mu \pm 3\sigma$ range can be considered an EMF, where μ is the average of number Γ collected from feedback ratings against a specific power node and $\sigma = \sqrt{\sum_{m=1}^{\Gamma}(TV_m - \mu)^2}$ is the standard deviation. The non-parametric cumulative sum (CUSUM) algorithm can be adopted for the detection of extreme malicious feedbacks (EMFs) within $\mu \pm 3\sigma$ [56].

Taking into account the social characteristics (such as interaction with other EVs, feedbacks from the use of the system, personal recommendations) and rating bias, the second phase is devoted to the obtention of the credibility of each normal feedback. Typically, an EV_i user tends to assign a greater trust value to their friends, whereas strangers start attributing values from average ones, which can be increased or decreased, according to the evolution of the interactions. Additionally, users´ recommendations become more reliable if the deviation between their ratings and others users´ rating on a certain power node is decreased.

The trust between system entities can be described using the graph $\Im$ where nodes represent EVs and edges represent the relationship $sr_{i,i'}$ involved in a pair of EVs (EV_i and $EV_{i'}$). The edge $sr_{i,i'} = 0$ when the EVs have no relationship, and $sr_{i,i'} = 1$ when they have close relationship [56]. Next, in figure 6, the vertices and edges related to an EV_i are shown.

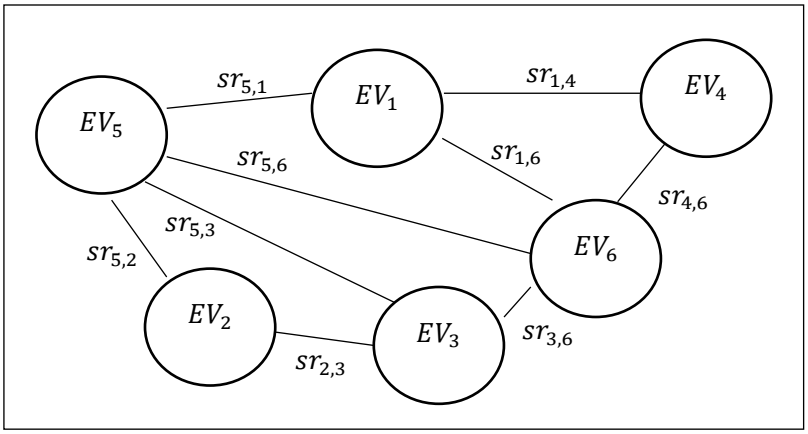


FIGURE 7: TRUST BETWEEN SYSTEM ENTITIES

Therefore, user EV_i's trust in user EV_i can be obtained by

$$ET_{i,i'} = sr_{i,i'} + \delta_s \sum_{\hat{\imath} \epsilon N_i} sr_{i,\hat{\imath}} \, . \, sr_{\hat{\imath},i'} \, . \tag{36}$$

where $\delta s \in [0,1]$ is a tuning parameter. N_i is the set of neighbors of EV_i, i.e., $sr_{i,\hat{\imath}} \neq 0$, $\forall \hat{\imath} \in N_i$. The deviation value of $EV_{i'}$ in entity j is defined by

$$div_{i',\mathrm{j}} = \frac{1}{H_{i',j}} \sum_{h=1}^{H_{i',\mathrm{j}}} |TV_{i',\mathrm{j}}^{h} - \overline{TV_j}| \, , \tag{37}$$

where $\overline{TV_j} = \frac{1}{\mathbb{I}_j} \sum_{i^{\sim} \epsilon \mathbb{I}_j} \frac{1}{H_{i',j}} \sum_{h=1}^{H_{i',\mathrm{j}}} TV_{i^{\sim},j}^{h}$ is the average rating of the recommendations obtained from information stored in the off-chain given by the $\boldsymbol{i}$ (EVs) that interacted with $\boldsymbol{j}$ (FS) and granted EV_i recommendation request and $H_{i,j}$ is the total number of interactions between the EV_i and entity FS. Therefore, for $EV_{i'}$, the credibility of the recommendation of EV_i to j is

$$cre_{i,i',j} = ET_{i,i'}\left(1 - div_{i',j}\right). \tag{38}$$

The implicit trust between the EV_i and entity j can be expressed by:

$$ITV_{i,j} = \frac{\sum_{i' \epsilon \mathbb{I}_j} \sum_{h=1}^{H_{i',j}} cre_{i,i',j} \, . \, TV_{i,j}^{h} \, \Phi(th)}{\sum_{i' \epsilon \mathbb{I}_j} \sum_{h=1}^{H_{i',\mathrm{j}}} cre_{i,i',\mathrm{j}} \, . \, \Phi(th)} . \tag{39}$$

Global Trust: The global trust of EV_i to j is the sum of implicit and explicit trust, i.e.,

$$GTV_{i,j} = \lambda_{i,j} \, ET_{i,i'} \left(1 - \lambda_{i,j}\right) . IT_{i,j} \, . \tag{40}$$

$\lambda_{i,j}$ is a weight value, denoted by $\lambda_{i,j} = \frac{H_{i,j}}{H_{i,j}+\delta_t}$, where $\delta t \in [0, 1]$ is a tuning parameter. Therefore, to increase the accuracy and weight of explicit trust, the interaction number should be higher.

2.6. Summary

This chapter described cryptographic techniques, security tools, security properties and cyber-attacks that can affect the system and a computational trust approach to be used in protocol proposals. The cryptographic techniques were chosen towards offering the proposed protocols security and high performance.

Diffie Hellman Key Agreement based on ECC, Short Signatures and Blind Signatures, bilinear pairing, and Hash Chain were used for the first protocol, whereas Key Agreement based on chaotic cryptography, signatures based on chaotic maps, and hash chain were employed for the second one. The third protocol is based on blockchain and used a key agreement founded on chaotic maps and hash chain. Finally, the fourth protocol was designed considering computational trust based on blockchain, Short Signatures, Blind Signatures, and bilinear pairing.

3. RELATED WORK

In this chapter, a literature review is presented, aiming to describe the relationship of each work to the others under consideration and to reveal any gaps that exist in the literature. The related works covered are divided into three groups: one that focuses on proposals for authentication protocols among various entities to protect against computer attacks and preserve privacy in dynamic charging systems; another related to the use of blockchain to ensure security of user information in VANET networks and, finally, a group that involves references related to computational trust, Smart Grids and VANETs.

3.1. Authentication of EVs in a Dynamic Charging System

Ensuring security and confidentiality in EV authentication and access control with the CWD-WPT system is one of the major concerns of the service. The user requires that their information not be stored on the devices in the middle of the connection between the EV and the charging station.

Several protocols have been proposed to authenticate a moving EV in a dynamic charging system. Among these protocols are Li et al. [57], Hussain et al.[23], Gunukula et al.[24], Rabieh and Wei [25], Pazos-Revilla et al. [58]; these protocols were proposed on an architecture that does not consider the utilization of the cloud to support the charging system. Another one (Tajmohammadi et al. [59]) proposes an authentication protocol with cloud architecture but does not carry out a formal security check.

Li et al. [57] presented an authentication protocol called “Fast Authentication for Dynamic EV Charging (FADEC)”, which has a dedicated short range communication (DSRC) based on the IEEE 802.11p standard and a five element architecture, i.e., the utility in charge of the management and administration of the CWD system, a Certification Authority (CA) that certifies all system keys, a set of pads installed on the highway for inducing energy to EVs, RSUs, which are wireless communication devices distributed over the sidewalk and interconnected through a backbone network, and EVs equipped with On board Units (OBU) that use dedicated short-range communication (DSRC) to communicate with RSUs.

The authentication protocol was based on the hash-based message authentication code (HMAC), which authenticates entities that rely on a symmetric key shared between two parties, the Elliptic Curve Digital Signature Algorithm (ECDSA), which authenticates vehicle safety messages, and Just Fast Keying (JFK), a key exchange protocol based on the Diffie Hellman protocol. Li et al. [57] do not emphasize the authentication process and establishment of the session key (JFK protocol). The security based on the JFK protocol has some flaws, since it does not protect the privacy of the user and is susceptible to repetition attacks.

Hussain et al.[23] designed a mutual authentication protocol that ensures privacy for a CWD system via charging plates (CPLs) installed under boards. The authors adopted the concept of online electric vehicle (OLEV) used in South Korea to name vehicles that receive an electric charge from the power line installed below the road surface. The network is based on a typical VANET consisting of EVs equipped with an OBU to communicate via DSRC with the infrastructure and a tamperproof module (TRM) that stores the EV´s confidential information. On the other hand, CPLs are installed on the track for recharging the EVs. The VANET Authority registers and revokes the system and the Tariff Service Provider Authority (CSPA) supplies energy to the CPs. The Department of Motor Vehicles (DMV) is at the top of the hierarchy, where each VANET Authority must be registered.

The protocol of Hussain et al. [23] uses the following cryptographic primitives to ensure protocol security: El Gamal encryption algorithm over elliptic curve cryptography (ECC), hash, hash chain, and XOR functions, for security analysis, which prove the resistance of the protocol against replaying attacks and impersonation, and dispute resolutions between EVs and the charging system. They have focused only on efforts to ensure mutual authentication and have not analyzed other security issues that may affect the system, such as integrity, DoS attack, Man In the Middle attack, amongst others.

Gunukula et al.[24] designed a protocol that preserves the security of the dynamic charging system and payment of the service. The network model considered in Gunukula et al.[24] is composed of a bank responsible for the sales of charging coins and verification of the validity of currencies. A charging service provider (CSP) manages the RSU group that is part of the charging station, the RSUs responsible for the management of the group of charging pads installed on the highway, and the charging pads responsible for the induction of energy to the EV.

Towards guaranteeing the security of the system, the protocol was based on the following cryptographic primitives: ECC-based partial blind signature, Diffie Hellman key agreement based on ECC, Exclusive OR and modified hash chain. The safety analysis describes the protocol of Gunukula et al. [24], which guarantees the anonymous authentication of the EV prior to the charging and disassociation of the EV with the currencies purchased. It also provides a description of resistance to attacks such as double spending, man in the middle, and others related to payment for the service; however, it does not analyse attacks that can affect the overall system.

Rabieh and Wei [25] proposed an efficient authentication protocol that guarantees the privacy of drivers. It is composed of EVs that use the charging system, and a charging management center (CMC), i.e., the main component of the architecture, controls the charging controllers and the charging pad (CP). The CPs are installed under the road and induce electric charge to the EVs. A charging controller is installed next to the highway and interconnects the CMC and the pads of the charging station. Finally, the charging carrier implements the necessary infrastructure for charging the EVs (CMC, charging controllers and CPs).

The protocol guarantees the security of client information through the following cryptographic primitives: hash chain, hash, Exclusive OR operations and blind signatures based on bilinear pairing. The security analysis describes the way the protocol performs a mutual authentication between the EVs and the system and guarantees the privacy of the EVs, unlinkability, double spending and anonymity of the EVs. Differently from other protocols, the one designed by Rabieh and Wei [25] considers an specific architecture of VANET and the security analysis does not consider several attacks that can affect the system such as injection, known key and impersonation attacks, among others.

Li et al. [60] developed an authentication protocol for a CWD-WPT station called “Portunes+”. The cryptographic primitives used for its creation are hash, AES encryption and signatures, and ECC-based subscriptions enhanced with Portunes +. The architecture involves a charging service provider (CSP), charging pads (CP) installed on the floor of the road, a pad owner located near the wheel and that controls the charging pad, and possible EVs that require charging. The protocol guarantees anonymity, integrity, and mutual authentication and resists attacks such as replay and impersonation; however, the authors did not discuss other attacks that might affect the system (e.g., privileged insider, known key, injection, random number leakage, injection, among others).

Tajmohammadi et al. [59] designed a cloud-based protocol for authentication and payment of load services at CWD stations supported by 5G networks. It uses symmetric keys established to guarantee the confidentiality of private information during message exchange, and low cost cryptographic primitives,

such as Hash and XOR functions to authenticate the EV. The architecture is composed of entities allocated in the cloud (Key Distribution Center (KDC), an Energy Provider (EP), a Bank and Trusted authority (TA), and entities allocated on the highway (RSUs, pads and EVs). The protocol guarantees privacy, anonymity, and mutual authentication and resists attacks such as impersonation, Man in the middle, replay, unlinkability, double spending, security against free riders, and privileged insider. However, the authors do not describe the way it guarantees integrity and resists password guessing, random number leakage, masquerade, and other attacks.

As can be seen in the above mentioned related works, although cloud computing has been dealt with in EV charging systems in VANET networks, fog computing has not yet been deeply explored for such systems.

Among the several advantages offered by fog computing over the traditional cloud are latency reduction and greater bandwidth. Regarding security analysis, the related studies investigated some attacks, but disregarded others that jeopardize the privacy, confidentiality, and availability of the system. This manuscript introduces an authentication and access control protocol that considers a fog-based system and utilizes Chebyshev chaotic maps, which most probably have not been used for CWD-WPT systems, to reduce computational and energy costs and improve security aspects. Resistance to different types of attacks (including some not treated in the related studies) is discussed and evaluated.

Table 1 shows a comparison of the entities and cryptographic primitives adopted by the protocols focused on CWD-WPT.

Table 1. COMPARISON AMONG ENTITIES AND PRIMITIVES

Protocol	Entities Considered		Cryptographic primitives	Cloud-based?	Formal verification Security?	Comparison with other protocols?
[57]	EV, pad, RSU, Utility, CA	5 entities	HMAC, symmetric key; ECDSA and Just Fast Keying (JFK)	Not	Not	Not
[23]	EV, CP, CSPA, VANET Authority	4 entities	ElGamal over ECC, hash, hash chain, and XOR functions.	Not	Not	Not
[24]	EV, CSP, RSU, pad, Bank	5 entities	ECC-based partial blind signature, Diffie Hellman key agreement based on ECC, XOR and modified hash chain	Not	Not	Not
[25]	EV, pad, C Company, CMC, C controller.	5 Entities	hash chain, hash, Exclusive OR operations and blind signatures based on bilinear pairing	Not	Not	Not
[60]	EV, CSP, RSU, pad Owner, Charging Pad	5 Entities	MACs, AES encryption, Hash, Portunes+ signature.	Not	Not	Not
[59]	Key Distribution Center (KDC), Energy Provider (EP), Bank, Trusted authority (TA), RSU, EV, Charging Plate (CP)	6 Entities	Symmetric key, Hash and XOR functions.	Yes	Not	Yes
Proposed Protocol PROT_1	EV, pad, RSU, Fog Server, Cloud(CCS)	5 Entities	Diffie Hellman Key Agreement based on ECC, Short Signatures and Blind signatures, bilinear pairing and Hash Chain.	Yes	Yes	Yes
Proposed Protocol PROT_2	EV, pad, RSU, FS, Cloud (CCS)	5 Entities	Key Agreement based on chaos cryptography, chaos-based signatures, and hash chain.	Yes	Yes	Yes

After carrying out an analysis of the related works, our proposal aims to address the security issues by performing an analytical and formal verification of a cloud-based CWD-WPT charging station.

3.2. VANET Security based on Blockchain

Several works have been published about security of VANET networks using concepts and resources of Blockchain.

Pazos-Revilla et al. [58] proposed a protocol for controlling access to a CDW WPT download station using anonymous authentication based on cryptographic primitives such as exclusive XOR and modified hash string and Diffie Hellman based on ECC. The architecture considers the owner of the EV who

wishes to use the dynamic charging station must purchase the ticket in a trusted bank and maintain communications with the service provider. Towards validating the authenticity of the currency, only the charging service provider (CSP) connects with the bank, thus avoiding exposing the user's identity. After the coin has been successfully validated, the EV goes to the charging station, which is comprised of a set of RSUs and manages a group of pads that charge the EV by induction. The authors analyzed the protocol resistance against attacks such as man in the middle and double spending, but did not analyze, for instance, masquerade, Forward secrecy, impersonation, privileged insider, random number leakage, injection, DoS, and known key, which might affect the system.

Jiang et al. [61] designed a distributed and secure wireless energy transfer architecture using blockchain for IoTs, including vehicles. Two types of plans, namely energy plan and Blockchain plan are considered and the architecture is composed of Smart devices (SD), mobile energy transmitters (MET), service station (SS), data access point (DAP), and miners. The authors employed a Blockchain consortium (hybrid), a DPoS consensus scheme, and an elliptic curve-based encryption scheme for its development. However, no security analysis was conducted and the way the scheme can resist computer attacks that can affect the system is not addressed.

Kim et al. [62] developed a static safe charging system for electric vehicles based on blockchain. The architecture is composed of an operator, several energy aggregators, and the EV. The authors used Blockchain and the basic concept of Hyperledger, whose efforts are focused on improving the performance and reliability of the bounty free blockchain which can be categorized as a private blockchain. The system uses Redundant Byzantine Fault Tolerance (RBFT) consensus, which is a variant of BFT and ECC as a scheme of encryption. Regarding security analysis, the authors included an analysis of attacks such as Replay and Impersonation, but did not consider others (e.g., MitM, DoS, masking, and injection).

Xu et al. [63] proposed a dynamic group key agreement protocol with Blockchain-based authentication that guarantees forward and backward secrecy and achieved great performance. The architecture consists of a key distribution center (KDC), general nodes (GN) that can be any type of device, and a private Blockchain with PoS consensus algorithm and bilinear pairing as a cryptographic scheme.

The security analysis included a description of the way the protocol resists attacks such as replay, impersonation, perfect forward secrecy, and perfect backward secrecy; however, other attacks that can affect the system (e.g., MitM, DoS, masking, and injection) were not considered.

Tan et al. [64] designed an authentication and key management scheme with no certificates for vehicles in a new model of VANET networks. The architecture is composed of a Trust Authority (TA), access points (AP), RSUs, and Vehicles. In the work [64], a consortium Blockchain system (hybrid) is used to create a group key of several vehicles, the encryption scheme used is based on bilinear pairing. On the other hand, the consensus method is not described. This work ([64]) does not describe how the proposed protocol resists attacks such as injection, MitM, known key, among others.

L. Roman and P. Gondim [65] proposed an authentication protocol for a CWD-WPT charging station based on bilinear pairing. the architecture proposed in this work consists of a control center server located in the cloud (CCS) that manages the FS, several fog servers (FS) that manage the RSUs of the charging station, each of which groups a number of charging pads and the charging pads embedded in a row on a lane of the highway. A security analysis of how the protocol resists different computer attacks was conducted and AVISPA successfully performed a formal security verification. The protocol does not include blockchain or trust management.

Below is a comparative Table 2 of the main characteristics of the studies that used Blockchain.

TABLE 2. COMPARISON OF PROPOSALS FOR BLOCKCHAIN-BASED SECURITY VANETS

Protocol	**Entities Considered**		**Cryptographic primitives**	**Type of Blockchain**	**Consensus**
[58]	EV, CSP, RSU, pad, Bank	5 Entities	hash chain. bilinear pairing, ECC	Not	Not
[61]	SD, MET, SS, DAP, and miners.	5 entities	ECC and Hash	consortium (hybrid)	DPoS
[62]	operator, energy aggregators, and EVs	3 entities	ECC and Hash	private	RBFT
[63]	KDC, GN, and devices	3 entities	bilinear pairing and hash	private	PoS
[64]	TA, AP, RSU, Vehicles	5 entities	TA, AP, RSUs and Vehicles	consortium (hybrid)	not described
[65]	EV, pad, RSU, Fog Server, Cloud(CCS)	5 Entities	ECC, bilinear pairing, Hash Chain.	Not	Not
Proposed Protocol PROT_3	EV, TA, FS, RSU, CCC, Charging Pad	6 entities	Chaos cryptography and Hash	consortium (hybrid)	RBFT

3.3. Computational trust in VANETs, Smart Grid and CWD-WPT

Below is a description of recently published studies on CWD-WPT charging systems and techniques related to computational trust for VANET and Smart Grid networks, and CWD-WPT systems.

K. Mannix et al. [54] discussed the general structure of a trust model, environments, types of attacks that violate both confidentiality and privacy of data, and the suitability of parameters and calculation methods according to environments and network types. The authors compared two trust models designed in two different Industry 4.0 networks. Such a study is relevant, since it demonstrated the design of a trust model depends on the service and the network architecture considered.

k. Hamouid and k. Adi [66] , proposed an anonymous authentication scheme for CWD-WPT charging for EVs, called FLPA (Fast and Lightweight Privacyaware Authentication), comprised of a Registration

Trusted Authority (RTA), a CSP, several pads, a bank, several RSUs, and EVs. The cryptographic scheme is based on bilinear pairing. The authors conducted an analysis and a comparison of communication and computation costs in relation to other protocols; however, no security analysis was performed.

W. Ahmed et al. [67] proposed a blockchain-based trust model and incentives to get EVs to participate in validating road events on VANET networks. The network model considered is composed of a TA, a public Blockchain system and PBFT consensus, RSUs and vehicles. The encryption scheme used is based on an elliptic curve. In this work, the following security properties and attacks are analytically analyzed: Privacy preservation, Unforgeability, Message authentication and reliability, False message attack, Sybil attack, Replay attack, Defense against Byzantine RSUs, On-off attack, Collusion attack. Finally, the authors compare the computational performance between the proposed system and the system proposed by other authors. Although it is a work that uses computational trust in VANETs, it addresses a specific work for trust in the road alert message service. Our work is focused on the CWD-WPT recharge service for EVs.

F. Ghajar et al. [55] designed a Scalable Blockchain Trust Management System (SBTMS) to support a Blockchain-based VANET and solve centralized problems and mutual distrust between VANET units. The vehicles use Bayesian formula and other blockchain information for checking the validity of the received message, so that later the reliability rate for each message. Vehicles carry the calculated rates to the RSUs to calculate the net reliability value. The authors used sharding consensus algorithm, but did not report on the type of blockchain adopted in the system. The network model considered only RSUs and no performance analysis of the vehicles was conducted. A security analysis considered false messages and no other types of attacks that might affect the system (e.g., replay, impersonation, MiTM, among others).

X. Wu et al. [68] proposed a lightweight and secure management scheme for a Harvesting-Dynamic Wireless Charging (EH-DWC) system that guarantees its effective authentication, secure communication, privacy protection, and reliable payment. The model is comprised of a TA, a power Supply Station (PSS), a Blockchain network, several RSUs, and EVs and the cryptographic scheme used is based on elliptic curve and bargaining game. However, no information on the blockchain used (class and consensus algorithm) is provided. A security analysis conducted considered the following characteristics of security and attacks: mutual authentication, secure communication, anonymity, and replay and MiTM attacks; however, it disregarded other types of attacks that might affect the system (e.g., Masquerade, Random number leakage, Privileged insider, among others). No performance comparison with other similar protocols was not performed.

T. Bianchi et al. [69] developed an authentication protocol for a Dynamic Wireless Power Transfer (DWPT) charging system, which is resistant to tracking of user´s activity and provides higher efficiency compared to authentication protocols. It contains the following entities: a Charging Service Provider Authority (CSPA), a vehicle registration and revocation authority, several charging pads, and EVs, and was designed with unique OR operations, hashing, and hashing chains. A security analysis was analytically performed for the following security properties and attacks: impersonation, unlinkability, MiTM, Free-riding, and Double-spending; however, it disregarded other attacks such as Privileged insider, Resistance to password guessing, Replay and Injection, among others. On the other hand, a performance comparison with other protocols was conducted regarding computational and communication costs.

R. Khalid et al. [70] proposes a blockchain-based trust management method that improves cooperation and privacy for multi-agent systems (MAS) in a Smart Grid network. To improve cooperation between agents and encourage them to restore their trust, a strategy based on game theory called tit-3-for-tat

(T3FT) was developed. On the other hand, to improve the agents' cooperation and perform Blockchain block validation more efficiently, a proof-of-cooperation consensus protocol is proposed. The results show that the trust model proposed in this work has better results compared to other proposals. The system model considered is composed of a Blockchain system, the agents. The security analysis does not consider internal attacks that can be carried out by dishonest agents, but rather performs an analysis where privacy, verifiability, fairness and transparency of the system are considered, in addition to demonstrating resistance to bad-mouthing and on-off trust related.

K. Qureshi et al. [71] propose a trust evaluation model for smart grids (TEMSG) to ensure the secure aggregation of data from smart grids (SG) and smart cities. To collect trust data and estimate the information, the authors use machine learning methods, to then evaluate and verify the accuracy and reliability of the system. The model of the SG system considered is composed of several modules such as: management, communication, electrical generation, transmission, residential and commercial and electrical storage. The authors carry out a general analysis of how the system can guarantee the security of the SG, but do not detail how the proposed system can support some attacks such as replay, injection, among others.

K. Boateng et al. [72] performs a study and contextualization of the use of trust in the Smart Grid under the conceptual domains and priority areas of NIST, multi-agent systems and the formalization of derived trust. The authors propose a new substation-based trust model and a Modbus variant to detect final-phase attacks. The variation was tested in different scenarios using two public datasets. The proposed model detects attacks on datasets and their influence on the behavior of the trust model. according to the results, the authors believe that the proposed trust model can be the basis for the creation of new models that fit other systems, such as the charging systems for EVs CWD-WPT.

Y. Wang [56] proposes a safe and efficient CWD-WPT charging scheme for vehicular power grids (VENs). The system model considered is composed of a consortium-type Blockchain system (unspecified consensus algorithm), RSUs, charging/unchanging pads and EVs. The Blockchain is used for access control and trust management, the game theory for managing recharge schedules and, finally, a cooperative mode of energy transfer is proposed. the work does not perform a detailed security analysis where it is described how the proposed system can resist computer attacks (such as: replay attack, masking attack, DoS, among others). In this work, analyzes and comparisons with other proposals are performed.

Below is a comparative table (Table 3) of the main characteristics of the studies that used in CWD-WPT and trust management.

TABLE 3 COMPARISON OF THE MOST RELEVANT DATA FROM RELATED STUDIES

	Entities of the proposal	**CWD-WPT Service**	**Trust management**	**Blockchain**	**Encryption technique**	**Security Analysis**	**Cost-based comparison with other proposals?**
[56]	Blockchain, EV, Pad, RSU	Yes	Yes	Yes	ECC, RSA	Yes	Yes
[66]	RTA, CSP, pads, bank, RSUs, EVs	Yes	Yes	Not	bilinear pairing.	Not	Yes
[68]	PSS, Blockchain, EV, RSU	Yes	Not	Yes	ECC and bargaining game	Yes	Not
[69]	CSPA, RRA, pads, EV	Yes	Not	Not	OR operations, hashing, and hashing chains	Yes	Yes
[67]	TA, Blockchain, EVs	**Not**	**Yes**	**Yes**	ECC	Yes	**Yes**
[54]	--	Not	Yes	**Yes**	--	Yes	**Yes**
[55]	RSU	Not	Yes	Yes	--	Yes	--
[70]	Blockchain, Aggregator, prosumer, physical slayer	Not	Yes	Yes	--	Yes	Yes
[71]	management, communication, electrical generation, transmission, and residential, commercial, and electrical storage	Not	Yes	Yes	--	Yes	--
[72]	Smart Grids	Yes	Yes	--	--	--	--
Proposed Protocol PROT_4	EV, Pad, RSU, Fog Server, Cloud(CCS), TA	Yes	Yes	Yes	Diffie-Hellman Key Agreement based on ECC, Short Signatures and Blind signatures, bilinear pairing and Hash Chain.	Yes	Yes

3.4. Summary

In this chapter, the related work has been separated in three sections, the first section mentions some works related to dynamic charging systems, in addition to a brief description of the characteristics such as architecture used, cryptographic schemes used and security analysis performed. For this first section, a comparative table was made (Table 1) where it is clearly shown the differences between the related work and the Proposed Protocols PROT_1 and PROT_2.

On the other hand, a second section of related work was created, where some published works were considered, focused on the use of Blockchain to guarantee the security of the system. For each work, a

brief description of the characteristics such as architecture used, cryptographic schemes used, type of blockchain and consensus system used was made. Table 2 was created to better visualize the differences between the works based on the blockchain and the protocol PROT_3

Finally, a literature review on the application of computational trust to VANET and smart grid networks and CWD-WPT systems and a verification of the authentication protocols of the latest CWD-WPT systems were conducted. To better visualize the differences between works related to trust management and the PROT_4 protocol, Table 3 was created.

The next chapter is devoted to the problem formulation and the proposal and evaluation of the protocols PROT_1 and PROT_2

4. PROBLEM FORMULATION AND PROPOSALS FOR CENTRALIZED CWD-WPT CHARGING STATION

This chapter introduces two protocols for a network model considered centralized, since the service is managed from the cloud, enabling the implementation of several recharge station control servers (geographically distant) for the management of the system. Two protocols (PROT_1 and PROT_2) were designed for the architecture.

4.1. Centralized CWD-WPT charging station system model

Figure 8 shows the network model with company charging server (CCS) (located in a cloud), EVs and a CWD-WPT charging station. Each CWD-WPT charging station is comprised of a fog server, multiple RSUs and charging pads.

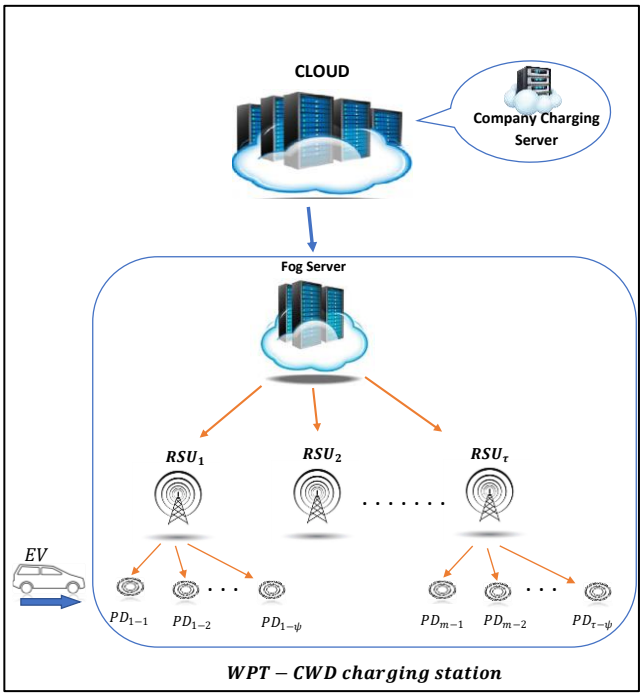


FIGURE 8. NETWORK MODEL WITH CENTRALIZED CWD-WPT CHARGING STATION

The system is assumed to have several CWD-WPT charging stations that communicate with the CCS. EVs can communicate with the CCS via the Internet. RSUs are access points installed on the roadside of the CWD-WPT charging station and can cover several kilometers.

We consider there are "$\boldsymbol{\tau}$" RSUs for one CWD-WPT charging station, and each RSU can communicate with a group of "$\boldsymbol{\psi}$" pads, while the fog server can communicate with all RSUs of the CWD-WPT charging station. Pads are elements that induce an electric charge to the EVs in motion using WPT. Each pad is activated through the validation of a unique key delivered by an EV. EVs can communicate with FS and RSUs through wireless networks, and with the pads through a short-range wireless communication device.

Towards a performance comparison among other protocols and the proposed scheme, according to [60] and [73], a charging station can be 4.2 km long and is managed by 1 FS (managed by CCS), 7 RSUs

(managed by FS) positioned 600 meters apart from each other, and each RSU manages 750 pads separated by 40 centimeters. Table 4 shows the characteristics of the charging station considered for this architecture.

Table 4**.** Characteristics of the charging station´s central architecture

Entities	FS	RSUs ($\boldsymbol{\tau}$)	Pads ($\boldsymbol{\psi}$) for 1 RSU
Number of entities for Charging Stations	1	7	750
Separation Between Entities of the Same Type	N/A	600 m	4cm

In the proposed protocols, FS is considered safe, RSUs and pads are considered safe but curious, and EVs are considered unsafe. On the other hand, communications that support the functioning of the system, but are not directly part of the authentication or access control processes, which are the focus of this thesis, are therefore assumed to be secure, that is, communications are secure between:

- the FS and the RSU, on all phases,
- RSUs and pads, on all phases,
- the EV and the CCS, on the phases of registration and purchase of tickets.

Communications are insecure between:

- the EVs and the FS in the authentication phase,
- the EVs and the RSU in the authentication phase,
- the EVs and pads in the authentication phase.

4.2. Adversary (attack) Model

Dolev Yao threat model [74] was used to analyze the security of the proposed protocol and the following assumptions were defined:

- The attacker can obtain any message from the network;
- An attacker can delete, spy, or modify messages transmitted over an insecure channel;
- An attacker can perform various attacks such as impersonation, Man in the middle, replay, unlikability, double spending, among others;
- Encrypted messages and hash functions are unbreakable.

For this system model, two protocols are proposed, considering the adversary model described in this section.

Protocols PROT_1 and PROT_2 are composed of 4 phases (see Figure 9) described in Sections 4.3 and 4.4.

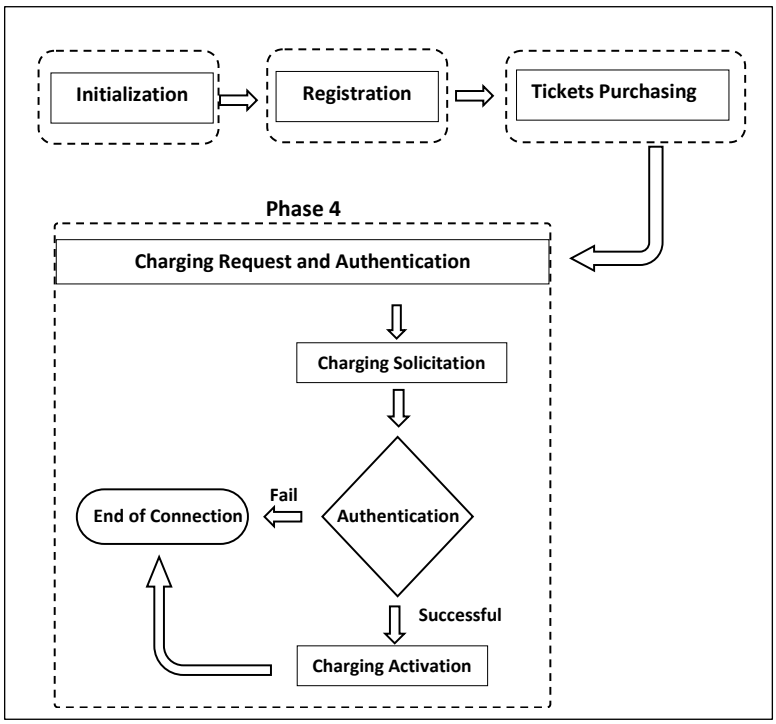


FIGURE 9. PHASES OF THE PROPOSED PROTOCOLS PROT_1 AND PROT_2

4.3. Protocol PROT_1 - Bilinear pairing-based authentication protocol for CWD-WPT charging system

The first protocol proposed for a CWD-WPT charging station is here described. This protocol was developed with cryptographic techniques based on bilinear pairing.

Our first protocol is divided into four phases, namely initialization, registration, ticket purchasing and charging request (see Fig. 8). In the initialization phase, sets, functions and master keys necessary for the start of the operation of the scheme are defined. In the registration phase, the data of the EV are stored in the system. In the phase of purchasing tickets, EVs purchase one or several tickets to perform the EV charge in the charging station. Finally, in the charging request phase, the delivery, validation, authentication and generation of keys necessary for the charging of EV through the CWD-WPT system are performed.

1st phase: **Initialization of the System**

In this phase, the use of the pseudorandom random number generator (PRNG) is considered for the generation of nonces and seeds. The PRNG will be reinitialized at random times, and the random value generated by the PRNG will be processed by a hash function to be used by the system. In PRNG, the initial state is changed with parameters that are the product of applying hash functions over input values concatenated with timestamps [51].

The system chooses two cyclic groups G_1 and G_2 of orders q and P and a generator element of group G_1 are chosen. G_1 and G_2 are supposedly related to a non degenerative pairing and a bilinear map that can be efficiently computed:

$\hat{e} : G_1 \times G_1 \rightarrow G_2$ such that $\hat{e}(P, P) \neq 1G_2$ and $\hat{e}(aP,bQ) = \hat{e}(b\,P,a\,Q) = \hat{e}(P_1, Q_1)^{ab} \in G_2$ for every $a, b \in Z_q^*$ and every $P, Q \in G_1$. Moreover, the hash functions of the system are defined: $H: \{0,1\}^* \rightarrow G_1$ and $H_1: \{0,1\}^*. G \rightarrow \mathbb{Z}_q^*$.

CCS then chooses a master private key $Y_{ccs}, \in Z_q^*$ and calculates its global public key $Y_{pub} = X_{ccs} * P$. Additionally, it computes its own public key $Q_{ccs} = H(ID_{ccs})$ and private key $S_{ccs} = X_{ccs} * Q_{ccs}$.

Finally, the CCS defines an elliptical curve on a finite field E (Fq) and parameters $\{G_1, G_2, \hat{e}$, P, H, $H_1, Y_{pub}, Q_{ccs}\}$ are published.

2nd phase: **EV registration**

All owners of EVs who want to use the CWD-WPT charging system register with the CCS through a secure channel. The user chooses a random number $X_{EV}, \in Z_q^*$ and calculates $Y_{EV} = X_{EV} * P$, where X_{EV} will be his/her private key and Y_{EV} will be the public key. This public key along with identity (ID_{EV}) and vehicle charging parameters (VCP) are sent to the CCS to be stored. Finally, the CCS creates a certificate $Cert_{EV} = X_{ccs} * Q_{EV}$ where $Q_{EV} = H(ID_{EV})$ and sends it to the EV.

3rd phase: **Tickets Purchasing**

Each ticket is assumed to have a specified amount of energy to be induced to the EV through a certain number of pads. The tickets are purchased through a secure channel and the EV has an associated bank account in the CCS, with enough money for their purchase.

The first message, m_1, requesting the purchase of n tickets to the CCS is sent by the EV.

$$m_1 = \{n, ID_{EV}, Cert_{EV}\}$$

The CCS receives it and generates n random values $\{r_1, r_2, \dots, r_n\} \in Z_q^*$. For each r_i for $0 \leq i \leq n$, $R_i = r_i * P$ is calculated and a message m_2 containing set R=$\{R_1, R_2, \dots, R_n\}$ is sent to the EV:

$$m_2 = \{R\}$$

The EV receives it, creates $\boldsymbol{n}$ random pseudonyms $\{PID_1, PID_2, \dots, PID_i \dots, PID_n\}$, and applies a blind signature to each $\boldsymbol{n}$ PID. It then chooses two random numbers $a, b \in Z_q^*$ and computes the blind pseudonym (B) for every pseudonym PID:

$$B_i = H(PID_i, \hat{e}(bQ_{ccs} + R_i + aP\ , Y_{pub}))+b \quad (41)$$

The EV sends message m_3 with the $B = \{B_1, B_2, \dots, B_i, \dots B_n\}$ to the CCS to receive the system signature.

$$m_3 = \{B\}$$

The CCS receives the message and signs all blind pseudonyms from set B:

$$Bs_i = (B_i * S_{ccs}) + (r_i * Y_{pub}) \quad (42)$$

It then sends message m_4 ($Bs = \{Bs_1, Bs_2, \dots, Bs_n\}$) to the EV.

Finally, the EV receives m_4 containing set Bs and calculates two values (J and L) for signature verification to obtain the signature of each blind pseudonym set $B = \{B_1, B_2, \ldots., B_i, \ldots. B_n\}$:

$J_i = Bs_i + aY_{pub}$, and $L_i = B_i - b$, therefore, the signature of each blind pseudonym B_i will be the pair of values (J_i, L_i). The figure 10 shows a summary of the ticket purchase phase and a summary of the ticket purchasing phase, respectively.

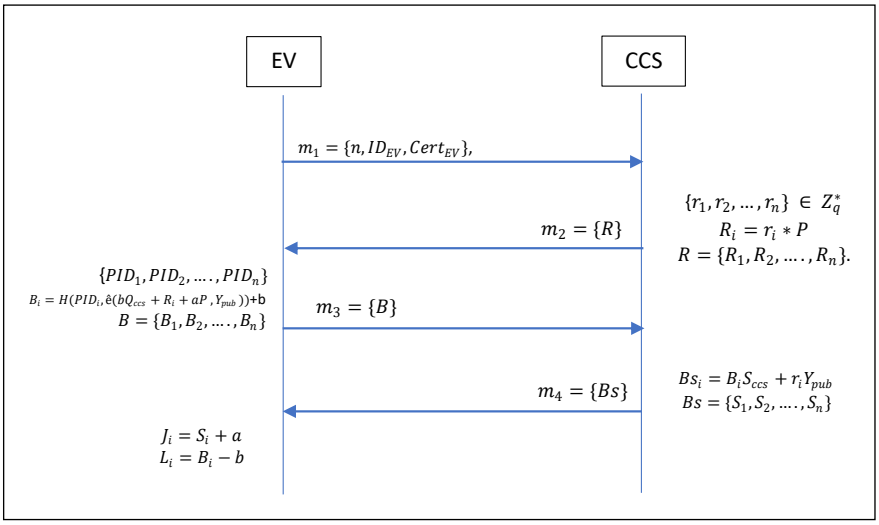


FIGURE 10. TICKET PURCHASING OF THE PROPOSED PROTOCOL PROT_1

4th phase: **Charging Request**

This phase describes the verification, authentication, and creation of session keys between the EV and the CWD-WPT charging station.

Once the EV owner has a valid ticket $(PID_1,\ J_1,\ L_1)$ and wants to charge his/her EV in a CWD-WPT charging station, the EV system selects a random number $\varphi_{EV} \in Z_q^*$, calculates $\phi_{EV} = \varphi_{EV} * P$, and sends an m_1 message to the fog server

$$m_1 = \{\phi_{EV}, t_5, H(\phi_{EV} || t_5)\}, \text{ where } t_5 \text{ is a timestamp.}$$

The fog server checks the hash and message timestamp m_1. If it succeeds, the server chooses a random value $\varphi_{fs} \in Z_q^*$ and calculates session $k_{fs-EV} = \varphi_{fs} * \phi_{EV}$ and values, such that the EV can calculate session key $\phi_{fs} = \varphi_{fs} * P$, verification key $VK = H(k_{fs-EV})$, and signature message $\sigma_{fs} = x_{fs} * H(\phi_{fs}, VK, t_5)$. The fog server immediately sends message m_2 to the EV.

$$m_2 = \{\phi_{fs}, VK, t_6, \sigma_{fs}\}$$

When $m_2' = \{\phi_{fs}', VK', t_6', \sigma_{fs}'\}$ arrives, the EV checks fog server's signature σ_{fs}': $\hat{e}(\sigma_{fs}', P) = ? \hat{e}(H(\phi_{fs}', VK', t_6'), Y_{fs})$. If the equality is successful, the EV authenticates the fog server, uses the message values to calculate session key $k_{fs-EV} = \varphi_{EV} * \phi_{fs}$, and verifies the integrity of the key calculating $VK = H(k_{fs-EV})$ and checking if $VK' = ? VK$. If the equality is successful, the EV uses the session key to crypt and send message m_3 containing the ticket (PID_1, J, L) and a timestamp to the fog server.

$$m_3 = \{PID_1, J_1, L_1, t_7\}_{k_{fs-EV}}$$

When the message arrives at the fog server, it is deciphered with session key k_{fs-EV}, the timestamp is checked and the pseudonym validity is immediately verified: $L_i = H(PID_i, \hat{e}(J_i, P)\hat{e}(Q_{ccs}, Y_{ccs})^{-L_i})$. If the validation is successful, the fog server chooses random seeds α_1, α_2, creates a new pseudonym $PID2_1 = H_1(PID_1 + \alpha_1)$, and sends an encrypted message m_4 containing seed α_1, τ and a timestamp to the EV. A message broadcast m_5 encrypted with key K_{G-RSU} and containing seeds α_1, α_2, τ and a timestamp is also sent to the group of RSUs. Finally, the fog server revokes pseudonym PID_1 to prevent its reuse.

$$m_4 = \{\alpha_1, \tau, PID2_1, t_8\}_{k_{fs-EV}} \text{ , sent to EV}$$

$$m_5 = \{\alpha_1, \alpha_2, \tau\ , PID2_1, t_9\}_{k_{G-RSU}}, \text{ sent to RSU}$$

When the EV receives m_4, it decrypts it and checks its timestamp. If the verification is successful, it calculates, offline, a verification key for each RSU using a hash chain $H^{RSU}(\alpha_1) = \{H(\alpha_1), H^2(\alpha_1), \dots H^{\tau}(\alpha_1)\}$. It also calculates, offline, and with each verification key, a message authentication code $HMAC^d_{RSU} = \{PID2_d||1||t_8||H^d(\alpha_1)\}$, and authenticates each RSU.

All RSUs receive the message m_5 from the fog server, decrypt with the group key (k_{RSU-G}) and check the timestamp. If the check succeeds, each RSU calculates a check key $H^d(\alpha_1)$, a session key $k_{RSU-PID2} = H(H^d(\alpha_1||d) \oplus H^d(\alpha_2))$, a verification key ($\boldsymbol{VK}$) and a message authentication code $HMAC^d_{RSU} = H(\ H^d(\alpha_2)||VK_2||t_{10}||H^d(\alpha_1))$, where $\boldsymbol{d}$ is the position of the RSU at the charging station d: 1≤d≤τ.

The authentication of the first RSU is explained in what follows for simplifying the description of the protocol. The authentication of the EV with the other RSUs and the group of pads managed by it undergo the same authentication process.

When the EV is authenticated with the first RSU, it sends a message m_6 containing message pseudonym $PID2_{EV}$, the sequence number of RSU, a timestamp, and an $HMAC^1_{RSU} = H(PID2_{EV}||1||t_9||H^1(\alpha_1))$.

$$m_6 = \{PID2_{EV}, 1, t_{10}, HMAC^1_{RSU}\}$$

When the message arrives, the RSU checks if its database contains PID_{EV}. If so, it checks $HMAC^1_{RSU}$ with the values associated with $PID2_{EV}$. If the verification is successful, the RSU computes session key $k_{RSU-EV} = H(H^1(\alpha_1||1) \oplus H^1(\alpha_2))$, and sends message m_7 containing a value $H^1(\alpha_2)$, a key verification code $VK_2 = H(k_{RSU-EV})$, and its signature $HMAC^1_{EV} = H(\ H^1(\alpha_2)||VK_2||t_{10}||H^1(\alpha_1))$ to the EV. It also adds the check key to a revocation list of RSUs to prevent reuse of the key.

$$m_7 = \{H^1(\alpha_2), VK_2, t_{11}, HMAC^1_{EV}\}$$

When $m_7' = \{H^1(\alpha_2)', VK_2', t_{10}', HMAC^1_{EV}'\}$ arrives, the EV checks the RSU's $MAC^1_{EV}' = ? HMAC^{\tau}_{EV} = H(H^1(\alpha_2)'||VK_2'||t_{10}'||H^1(\alpha_1)$. If the equality is successful, the EV authenticates the RSU and uses the message values to calculate session key $k_{RSU-EV}' = H(H^1(\alpha_1||1) \oplus H^1(\alpha_2)')$. It also verifies the integrity of the key calculating $K_2 = H(k_{rsu-EV}')$, and compares $VK_2' = ? VK_2$. If the equality is successful, the EV uses the session key to send an m_8 message containing a hash chain request to the RSU.

$$m_8 = \{\text{hash chain request}, t_{12}\}_{k_{rsu-EV}}$$

The RSU receives, decrypts, checks the timestamp (t_{12}), and sends message m_9 to the EV. ψ is the number of keys to be authenticated in each pad and $v \, \epsilon \, Z$ is a random number used as the initial value for the calculation of the hash chain. Additionally, the RSU sends all pads a message broadcast m_{10} encrypted with group key (k_{G-pad}) that contains public hash chain verification key $k_{PH} = H^{\psi+1}(v)$ used for the verification of the keys sent by EV.

$$m_9 = \{\psi\ , v, t_{13}\}_{k_{rsu-EV}}$$

$$m_{10} = \{k_{PH}, t_{13}\}_{K_{G-pad}}$$

The EV receives and decrypts m_9 with values ψ and v, and computes hash chain $H^{\psi}(v)$. Each block of pads managed by the RSU receives and decrypts broadcast message m_{10} with the group key. The message contains public hash chain verification key $k_{PH} = H^{\psi+1}(v)$. Whenever a key from a hash chain is sent by the EV (m_{11}) to one of the pads, the pad checks if the key has been validated by iteratively applying $\xi - \psi$ (for $0 \leq \xi \leq \psi + 1$) times the hash function and compares it to the public key hash chain (verification key). If the verification is successful, the pad checks the status of the key in the revocation list. If the key has not been revoked, it accepts the key sent by the EV and revokes it to avoid double use. The process ends when the EV has passed over all pads.

Below is the mathematical proof of the signing blind pseudonym and fog server's signature verification:

- Signing blind pseudonym verification:

$$L =? H(PID, \hat{e}(J, P)\hat{e}(Q_{ccs}, Y_{ccs})^{-L}), \quad (43)$$

$$L = H(PID, \hat{e}(J, P)\hat{e}(Q_{ccs}, Y_{ccs})^{-L}), \quad (44)$$

$$= H\left(PID, \hat{e}\left(B.S_{ccs} + r.Y_{pub} + a.Y_{pub}, P\right)\hat{e}(-L.Q_{ccs}, x_{ccs}.P)\right), \quad (45)$$

$$= H\left(PID, \hat{e}\left(B.S_{ccs} + r.Y_{pub} + a.Y_{pub}, P\right)\hat{e}(-(B-b).Q_{ccs}, x_{ccs}.P)\right), \quad (46)$$

$$= H\left(PID, \hat{e}\left(B.S_{ccs} + r.Y_{pub} + a.Y_{pub}, P\right)\hat{e}((-B+b)(Q_{ccs}.x_{ccs}), P)\right), \quad (47)$$

$$= H\left(PID, \hat{e}\left(B.S_{ccs} + r.Y_{pub} + a.Y_{pub}, -B.S_{ccs} + b.S_{ccs}, P\right)\right), \quad (48)$$

$$= H(PID, \hat{e}(r.Y_{pub} + a.Y_{pub} + b.(Q_{ccs} * x_{css}), P)), \quad (49)$$

$$= H\left(PID, \hat{e}\left(b.Q_{ccs} + R_i + a.P\ , Y_{pub}\right)\right). \quad (50)$$

- Fog server's signature verification: $\hat{e}\left(\sigma_{fs}', P\right) =? \hat{e}\left(H\left(\phi_{fs}', VK', t_5'\right), Y_{fs}\right)$.

$$\hat{e}\left(\sigma_{fs}', P\right) = \hat{e}\left(H\left(\phi_{fs}', VK', t_5'\right), Y_{fs}\right), \quad (51)$$

$$= \hat{e}\left(H\left(\phi_{fs}', VK', t_5'\right), x_{fs} * P\right), \quad (52)$$

$$= \hat{e}\left(x_{fs} * H\left(\phi_{fs}', VK', t_5'\right), P\right), \quad (53)$$

$$= \hat{e}\left(\sigma_{fs}', P\right). \quad (54)$$

Figure 11 shows the flow of messages exchanged among the entities in the charging request phase.

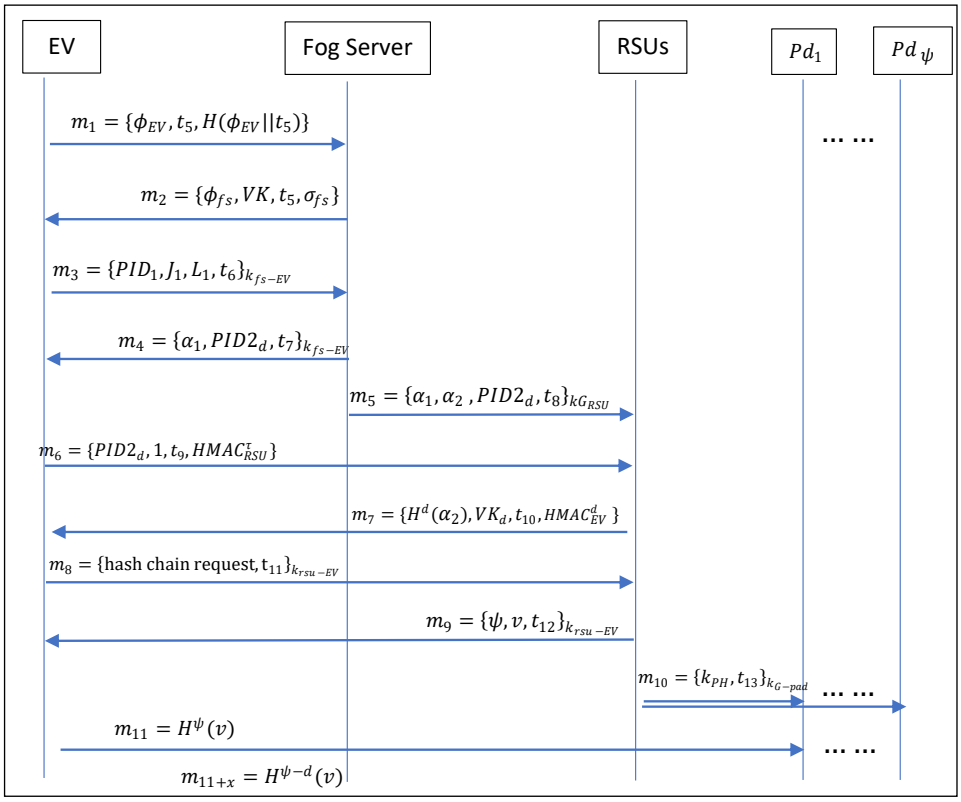


FIGURE 11. CHARGING REQUEST PHASE OF THE PROPOSED PROTOCOL PROT_1

4.3.1. Comparative Performance Evaluation

This subsection reports on a performance analysis of computational and communications costs. The authentication procedures between the fog server and EVs (FS EVs), EV and RSUs (EVs RSU), and EVs and pads (EVs pads) were assumed independent, since those processes can be conducted in different time periods and locations. For example, an EV can authenticate the fog server far from the charging station with considerable time in advance. The following EVs RSUs authentication process can be performed hundreds of meters from the first pad and several seconds in advance. Lastly, an EV must be authenticated by the pad a few centimeters from it and microseconds in advance.

4.3.1.1. Communication Costs

We consider that this transmission uses high coverage communication technology such as LTE, so that the EV is able to perform the exchange of information with the FS before entering the CWD-WPT charging station. For communications within the CWD-WPT charging station (EV RS and RSU PAT Communications) DSRC communications technology would be used which, within the effective communication range, has better communication performance than LTE. As in [75], the combination of DSRC and LTE has been considered a good solution for VANET.

Communication cost refers to the total number of bytes transmitted by a network during the execution of a protocol, without considering the headers or control bits inherent to the communication protocol used. Table 5 shows the values in bytes of each variable used. (Values taken from Rabieh and Wei [25]).

TABLE 5. SYMBOLS AND COSTS IN BYTES [25]

Symbol	Description	Length (Bytes)
ID	Identification	128
PID	Pseudo identity	32
$H(\,)$	Hash function	32
X	Private key	32
Y, Q	Public key	32
k	Session key	32
σ	Digital signature	32
(J, L)	Blind signature	96
ϕ	Pre key of session	32
τ	Number of RSUs per fog server	8
ψ	Number of pads per RSU	8
α, v	Seed	20
t	Timestamp	8
VK	Verification key	32
hash chain request	Hash chain request	8
*	Multiplication operator	-
$\hat{e}$	Bilinear Pairing	-
CCS	Company Charging Server	-
RSU	Roadside unit	-
HMAC	Hash-based message authentication code	32
P	Generator point of the elliptical curve	32

To calculate the communication costs using Table 4 of an EV that will authenticate to the fog server, the first RSU and the first pad, we have:

- $m_1 = \{\phi_{EV}, t_5, H(\phi_{EV}||t_5)\} = 32 + 8 + 32 = 72\ Bytes$
- $m_2 = \{\phi_{fs}, VK, t_6, \sigma_{fs}\} = 32 + 32 + 8 + 32 = 104\ \ Bytes$
- $m_3 = \{PID_1, J_1, L_1, t_7\}_{k_{fs-EV}} = 32 + 96 + 8 = 136\ \ Bytes$
- $m_4 = \{\alpha_1, \tau, PID2_1, t_8\}_{k_{fs-EV}} = 16 + 8 + 32 + 8 = 64\ Bytes$
- $m_5 = \{\alpha_1, \alpha_2, \tau\ , PID2_1, t_9\}_{k_{G-RSU}} = 16 + 16 + 8 + 32 + 8 = 80\ Bytes$
- $m_6 = \{PID2_{EV}, 1, t_{10}, HMAC^{\tau}_{RSU}\} = 32 + 8 + 8 + 32 = 80\ Bytes$
- $m_7 = \{H(\alpha_2)^{\tau}, VK_2, t_{11}, HMAC^{\tau}{}_{EV}\ \} = 32 + 32 + 8 + 32 = 104\ Bytes$
- $m_8 = \{\text{hash chain request}, \mathrm{t}_{12}\}_{k_{rsu-EV}} = 8 + 8 = 16\ Bytes$
- $m_9 = \{\psi\ , v, t_{13}\}_{k_{rsu-EV}} = 8 + 16 + 8 = 32\ Bytes$
- $m_{10} = \{k_{PH}, t_{13}\}_{K_{G-pad}} = 32 + 8 = 40\ Bytes$
- $m_{11} = \{H(v)^{\psi}\ \} = 32\ Bytes$

Table 6 shows the comparison of communication costs between the protocols proposed by Gunukula et al.[24], Rabieh et al.[25] and our protocol, counting the bytes (according to Table 2) of the messages

exchanged between entity pairs and the total number of messages exchanged by $\boldsymbol{n}$ EVs that try to enter the wireless charging system composed of $\boldsymbol{\tau}$ RSUs and $\boldsymbol{\psi}$ pads charging by RSUs.

TABLE 6. COMPARISON OF COMMUNICATION COSTS IN BYTES (PROPOSED PROTOCOL PROT_1)

Message	Gunukula et al.[24]	Rabieh et al.[25]	Proposed Protocol PROT 1
M1	$32n$	$224n$	$72n$
M2	$128n$	$248n$	$104n$
M3	$168n$	$128n$	$136n$
M4	$136n$	$128n$	$64n$
M5	$32(n*\tau)$	$40(n*\tau)$	80n
M6	$32(n*\tau)$	$40(n*\tau)$	$80(n*\tau)$
M7	$32(n*\tau)$	$32(n*\tau*\psi)$	$104(n*\tau)$
M8	$20(n*\tau)$	--	$16(n*\tau)$
M9	$32(n*\tau*\psi)$	--	$32(n*\tau)$
M10	--	--	$32n$
M11	--	--	$32(n*\tau*\psi)$
Total	$\boldsymbol{n\,(464+\tau\,(116+\,32\psi))}$	$\boldsymbol{n\,(728+\tau\,(80+\,32\psi))}$	$\boldsymbol{n\,(488+\tau\,(232+\,32\psi))}$

Figure 12 shows a comparison of the communication costs the protocols proposed in references [24], [25] and our protocol. The values adopted for evaluation of communication costs are based on Li et al. [60], who proposed parameters for the modeling of a typical CWD-WPT charging station. According to Table 4, the costs of the 3 (three) proposals are very similar; they can slightly differ in function of the values of *n (*EVs*)*, *τ(* 7 RSUs*),* and *ψ(*750/RSU*)*.

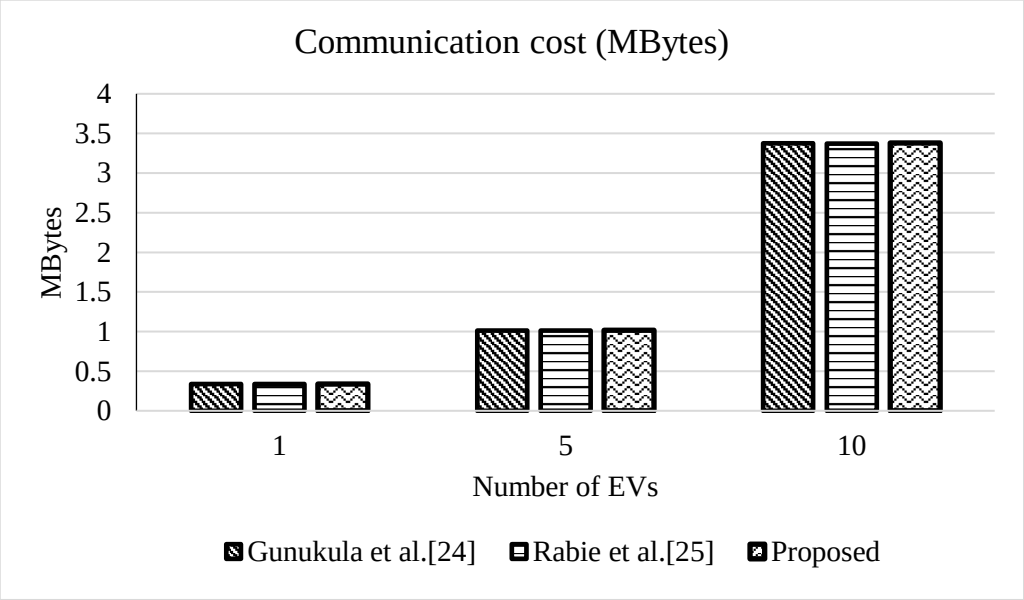


FIGURE 12. COMMUNICATION COSTS COMPARISON (PROPOSED PROTOCOL PROT_1)

4.3.1.2. Computational Costs

Below is the calculation of the computational costs of the entities of the network model. Table 7 shows the execution times of the Multiplication (T_{mul}), Exponentiation (T_{exp}) and Bilinear Pairing (T_{pair}) functions based on Tao et al. [52], for each entity. The execution costs of hash function, signature message, and message signature for RSU and FS were analytically calculated through an interpolation of the execution times characterized in τ. Therefore, 70% of the execution costs of the operations were taken (which is not responsibility of τ) for the definition of the RSU execution time, and 60% of the costs of their execution (which is not responsibility of τ) were used for the definition of the FS execution time.

The time costs of operations as symmetric encryption/decryption and addition, have been omitted, because their execution times are very short and rarely used in the protocol, in comparison to the Hash operation.

TABLE 7. COSTS IN $\boldsymbol{ms}$ OF EACH OPERATION AND ENTITY CONSIDERED FOR PROPOSED PROTOCOL PROT_1 (ADAPTED FROM [52])

Entity	Parameters of the entities involved			Costs (ms)					
	CPU(GHz)	RAM	OS	T_{mul}	T_{exp}	T_{pair}	T_{hash}	T_{g-sig}	T_{v-sig}
EV/Pad	Qualcomm(R) Octa core 1.5	2	Androi d 4.2.2	0.50	0.54	16.6	$0.043x10^3$	0.6	0.78
RSUs	Intel(R) Dual core 3.1	4	64 bit Win 7	0.36	0.38	11.5	$0.03x10^3$	0.42	0.55
CCS/CMC/FS	Intel(R) Hexa core 1.6	16	16 Win server 2012	0.3	0.31	8.6	$0.025x10^3$	0.36	0.47

In what follows is the calculation of the computational cost of each entity in the proposed protocol:

- Each EV executes 2 multiplications ($\boldsymbol{T_{mul}}$) to create the session key as FS, a 1 (one) verification of the FS signature ($\boldsymbol{T_{v-sig}}$), $\boldsymbol{\psi}$ hash ($\boldsymbol{T_{hash}}$) for each pad, 3 hashes ($\boldsymbol{T_{hash}}$) to authenticate FS process, and 2 hashes ($\boldsymbol{T_{hash}}$) to authentication RSU process.
- Each FS executes 2 multiplications ($\boldsymbol{T_{mul}}$) to create the session key as EV, 1 (one) signature ($\boldsymbol{T_{g-sig}}$), and 4 hashes ($\boldsymbol{T_{hash}}$), 1 Exponentiation ($\boldsymbol{T_{exp}}$), and Bilinear Pairing ($\boldsymbol{T_{pair}}$) to authenticate the EV process.
- Each RSU executes 4 hashes ($\boldsymbol{T_{hash}}$) for authenticating the EV process.
- Each Pads executes $\boldsymbol{\psi}$ hash ($\boldsymbol{T_{hash}}$) for authenticating the EV process.

Table 8 shows a comparison of the number of operations performed by the protocols of Gunukula et al.[24], Rabieh et al.[25] and the Proposed Protocol PROT_1. Like the other protocols, the proposed protocol performs the operations with higher computational costs in the entity with greater computational capacity (in our case the FS). On the other hand, entities with lower capacity such as EV, RSU, and pads perform less complex operations to ensure lower latency for the CWD-WPT scheme.

TABLE 8. COMPUTATIONAL COSTS COMPARISON (PROPOSED PROTOCOL PROT_1)

Protocols	**EV**	**CSP BNK/ CMC /FS**	**RSU**	**pad**
Gunukula et al.[24]	$2T_{exp}$ + $((\tau+1)^2+(\psi+5)T_{hash}$ +$1T_{v-sig}$	$2nT_{exp}$+$4nT_{mul}$+ +$((\tau+1)n)^2T_{hash}$ +$1nT_{g-sig}$ + $2nT_{pair}$	$(2n+((\tau+1)n)^2T_{hash}$	$n(\psi-1)T_{hash}$
Rabieh et al.[25]	$2T_{exp}$ + $(3+\psi)T_{hash}$ +$2T_{v-sig}$	$5nT_{exp}+4nT_{mul}$ $((3+\psi)\tau)nT_{hash}+$ $2nT_{g-sig}$ +$2nT_{pair}$	-------	$n(\psi)T_{hash}$
Proposed Protocol PROT_1	$2T_{mul}$ + $(5+\psi)T_{hash}$ + $1T_{v-sig}$	$2nT_{mul}$+ $1nT_{exp}+1nT_{g-sig}$ +$4nT_{hash}+2nT_{pair}$	$4nT_{hash}$	$n(\psi)T_{hash}$

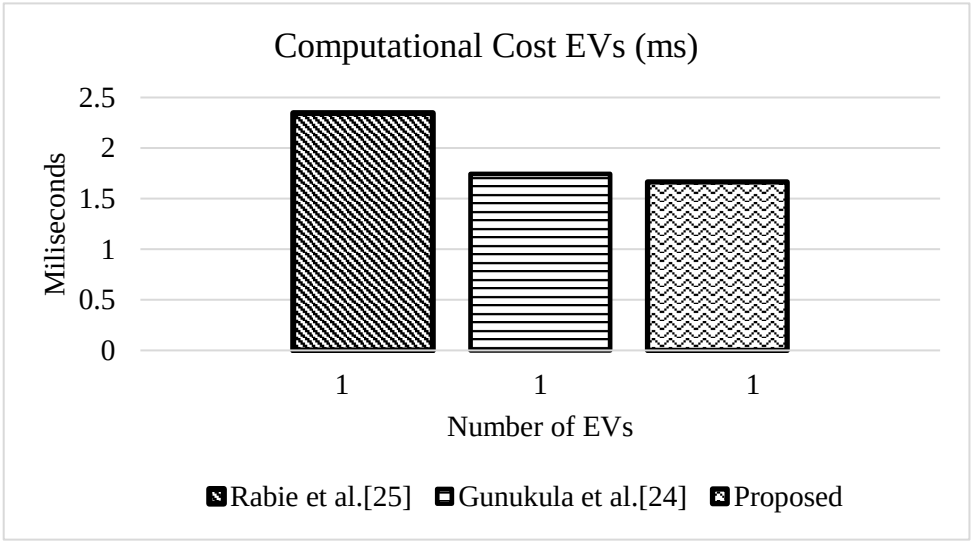


FIGURE 13.a COSTS OF EVS

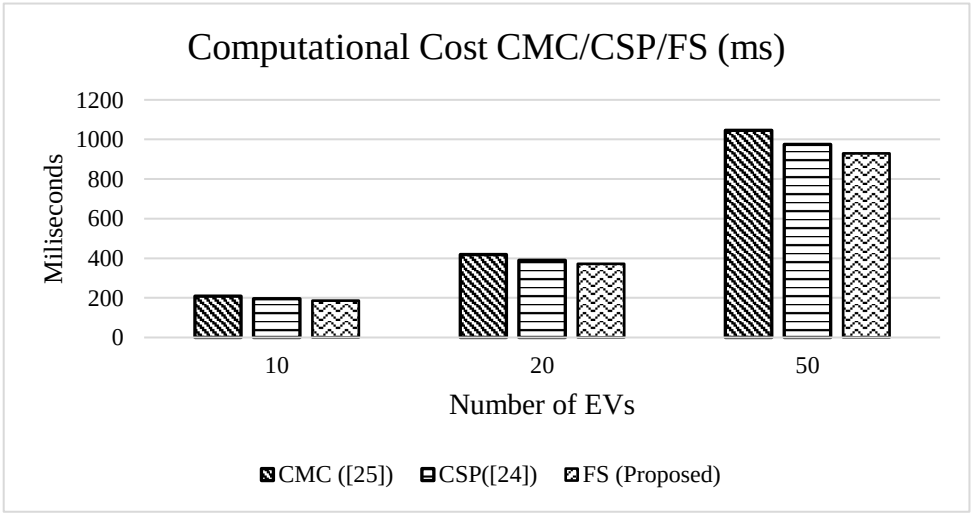


Figure 13.b Costs of CMC/CSP/FS

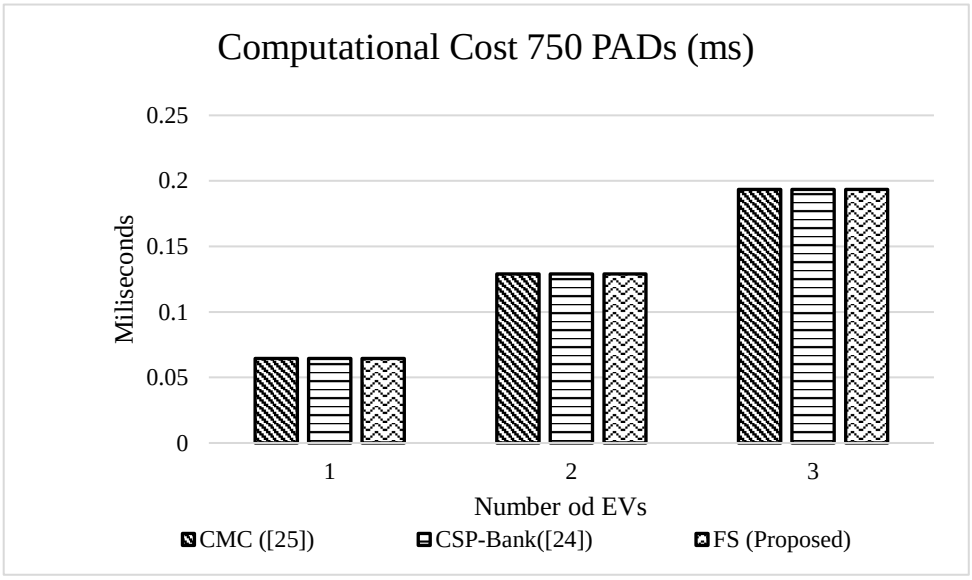


Figure 13.d Costs of pads

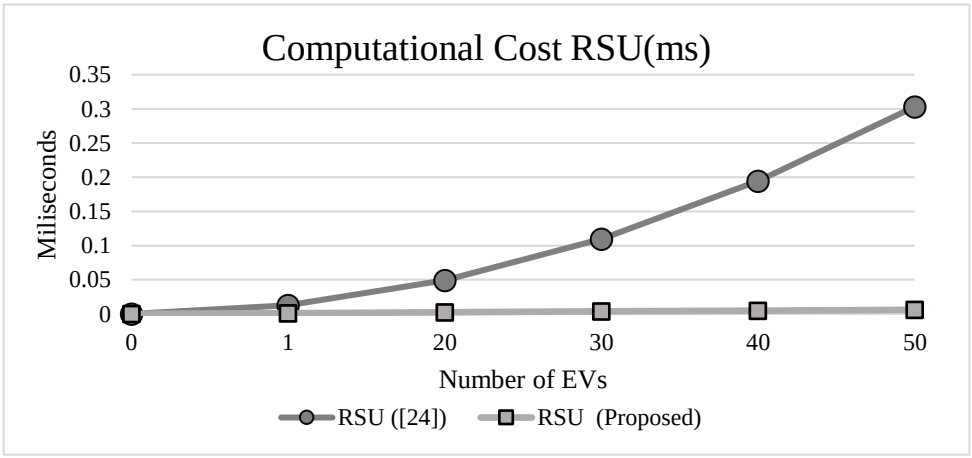


Figure.13.c Costs of RSUs

FIGURE 13. COMPUTATIONAL COSTS COMPARISON (PROPOSED PROTOCOL PROT_1)

In Figure 13 a comparison of the total computational costs of each entity is shown in the authentication phase of the protocols of Gunukula et al.[24], Rabieh et al.[25] and the Proposed Protocol PROT_1. The proposed protocol has a better computational cost for EVs, FS and RSU, and maintains the same computational costs of the other protocols for a group of 750 pads.

4.3.1.3. Energy Costs

The costs of the energy consumed in the execution of cryptographic operations in the protocols were compared. Equation $\boldsymbol{E_C} = T_{EX} * W$ (joules units),where T_{EX} is the execution time in ms and W is the maximum power CPU, calculated the energy costs. W = 10.88 watts [76][77] was assumed for the comparison of the energy costs of the proposed protocol with those of [24] and [25]. According to Figure 14, our protocol consumed the lowest energy.

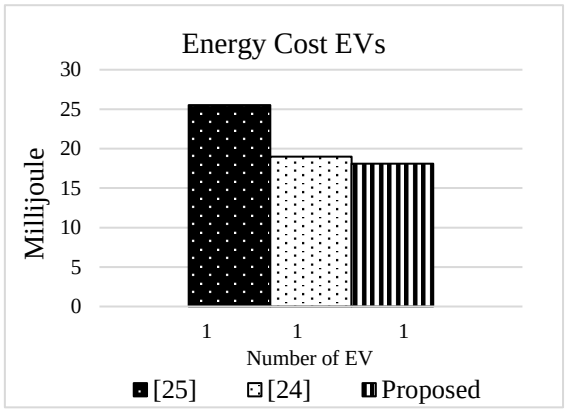


Figure 14.a Energy Costs of EVs

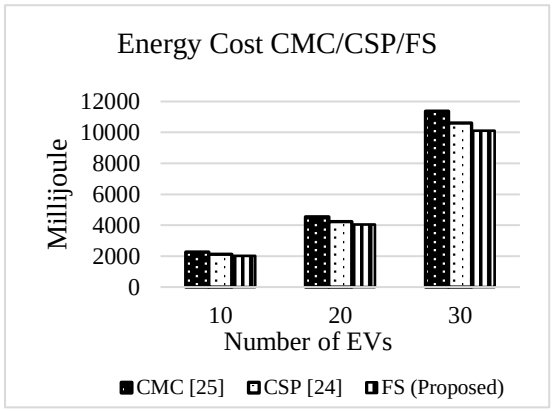


Figure 14.b Energy Costs of CMC/CSP/FS

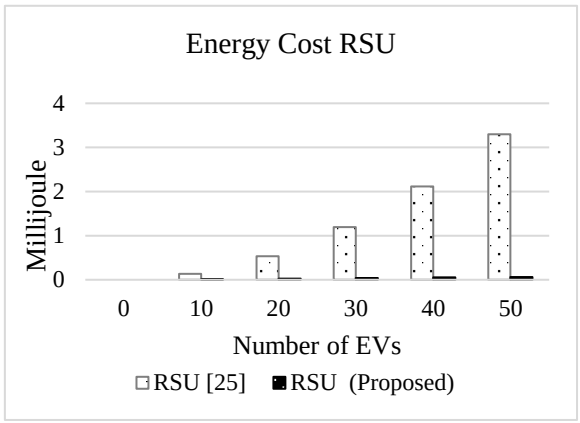


Figure 14.c Energy Costs of RSUs

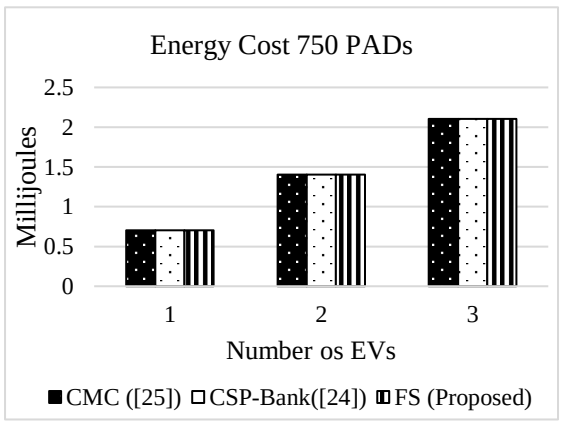


Figure 14.d Energy Costs of Pads

FIGURE 14. COMPARISON OF ENERGY COSTS (PROPOSED PROTOCOL PROT_1)

In comparison with other proposals, our scheme has yielded better computational and energy costs; it provides better results regarding security analysis and more complete results regarding safety analysis, and avoided problems related to centralization caused by the use of a cloud environment composed of fog computing and cloud computing. Such a combination promotes a better distribution of the computational processing of operations in the devices and guarantees lower latency in communications. Moreover, the protocol has met the security objectives, according to a formal verification conducted by AVISPA tool.

4.4. PROT_2 - Chaotic Maps based authentication protocol for CWD-WPT charging system

In this session, the second protocol proposed for a CWD-WPT charging station is described. This protocol was created considering the same system model, problem model and the same attack model considered in the first protocol.

The main differences with the first protocol are the cryptographic techniques based on Chaos cryptosystem, the management of a group with a binary tree of the system elements, the implementation of a fourth phase in the protocol and a calculation and comparison of the energy costs of the protocol. The proposed protocol comprehends the following four phases (Fig. 9):

- System Initialization: functions, properties, secret keys, and public keys are defined for the start of the system operation;

- Registration: the EV shares its data with the system, which validates and delivers some values to the EV to further identify itself;
- Ticket purchase: EVs acquire several tickets to be used at the charging station; and
- Charging Request and Authentication: the owner of the EV requests the reloading of the vehicle informing the ticket in the previous phase (Tickets purchase); the charging station authenticates the ticket. If the authentication fails, the system ends the connection; otherwise, it activates EV recharging and then ends the connection.

Nonces and seeds are generated by a pseudorandom number generator (PRNG) and values pre-processed by hash functions and concatenated with timestamps are used for the initial state of the generator. During its operation, the PRNG is reinitialized in random time periods and a hash function defined by the system processes the generated values, as recommended in [51].

The VANET infrastructure is assumed insecure ([16], [78]). The keys and parameters described below were created offline during system startup using chaotic encryption, but the step by step of their creation and distribution will be dealt with in another work. Each FS has a public key Y_{fs} calculated from a private key x_{fs}. The RSU is connected to the fog server and has a group key K_{G-RSU}, a private key x_{rsu} , and a public key Y_{rsu}. Otherwise, the pads and RSUs are connected and share a group key for pads K_{G-pads} to be defined. The use of groups improves the efficiency of the system, hence, security between the entities involved.

The protocol was created taking into account the definitions in section 2.4 , according to which chaos-based cryptography guarantees the security and confidentiality of information, since the results of Chebyshev chaotic maps operations satisfy the DLP (definition 2) and DHP (definition 3) properties.

1st phase: **System Initialization**

The system initialization phase considers a large prime number $\boldsymbol{p}$, the product $n = \overline{pq}$ of two primes $\bar{p}$ and $\bar{q}$ (taken as secret values of the system) and a factor of $p-1$, and β (a generator of the multiplicative group $\boldsymbol{G}$ and a member of an infinite group $GF\ (p)$ of order module $\boldsymbol{n}$ [48]). Given the function $\varphi(n) = (\bar{p}-1)(\bar{q}-1)$, the system selects a random number $\boldsymbol{e}\ \epsilon\ Z_p^*$ such that $gcd\ (e,n)\ =\ 1$

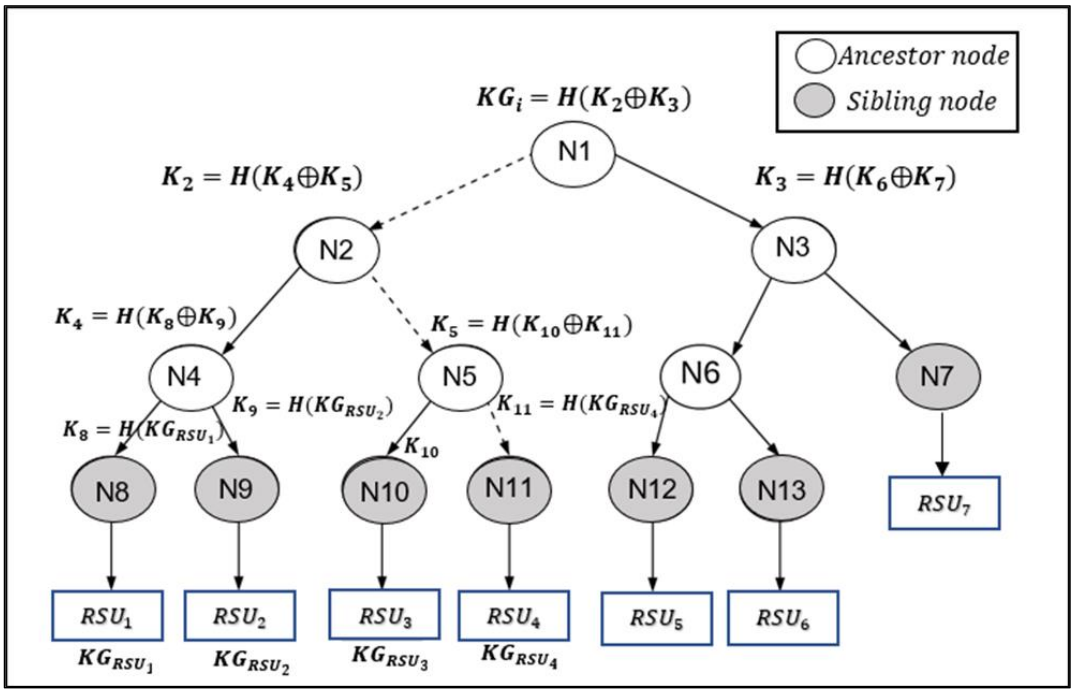


FIGURE 15. BINARY TREE WITH THE GROUP OF RSUS

and a number $\boldsymbol{d}$ satisfying $e.d = 1\ (mod\ \varphi(n))$. $H:\{0,1\}^* \rightarrow \mathbb{Z}_p^*$ is defined as the hash function of the system. The CCS selects a master private key $x_{ccs}, \in Z_q^*$ and creates its global public key $Y_{pub} = T_{x_{ccs}}(\beta) mod(p)$; $\boldsymbol{T}$ is a Chebyshev polynomial map of degree β defined by the following recurrent function[45]:

$$T_{x_{ccs}}(\beta) = \cos\left(x_{ccs} * arcos(\beta)\right) = \begin{cases} T_0(\beta) = 1\, mod(p),\ for\ x_{ccs} = 0; \\ T_1(\beta) = \beta\ mod(p),\ for\ x_{ccs} = 1; \\ T_{x_{css}}(\beta) = (2\rho\, T_{x_{ccs}-1}(\beta) - T_{x_{ccs}-2}(\beta))\ mod(p),\ for\ x_{ccs} > 1 \end{cases} \quad (55)$$

Additionally, the CCS calculates its own pair of public $Q_{ccs} = T_{H(ID_{ccs})}(\beta) mod(p)$ and private $S_{ccs} = T_{x_{ccs}}(Q_{ccs}) mod(p))$ keys, which must be altered in predefined periods of time.

The system creates a binary tree based on Parne et al. [79] with the group of RSUs that is part of the reload station. Figure 15 illustrates Binary tree with the group of RSUs.

In the group management schema created by FS, two leafs node are designated for each of last node in the tree (e.g. N4 node has leafs N8 and N9). The EVs and RSUs are associated with such leaf nodes, respectively, and the group signature (KG_i) is calculated on the root node. KG_i is used by group members to provide privacy protection and mutual authentication between EV and the charging station. The secret value of the Ki node is calculated by the entire N_i inner node in the binary tree as where left(i) and right(i) denote, respectively, the left and the right children of a Ni node. Function H is a hash function.

$$K_i = H(K_{left(i)}) \oplus H(K_{righ(i)})) \quad (56)$$

Ancestors, defined as the nodes in the path of leaf nodes (associated with group members) to the root node, form an ancestor set. Leaf nodes also have a set of siblings that are nodes born from the same node as the parents. Figure 15 shows the ancestor set and set of siblings of the N11 node (RSU_3.). Each group member maintains a private group subscription (KG_{EV_i} or KG_{RSU_i}.) and the associated node has a blind signature $H\left(KG_{EV_i}\right)$ or $H\ (KG_{RSU_i})$.

The Fog server in message $m_{initial}$ delivers each RSU a list of blind values of the set of sibling nodules and the nodule of the EV_i. For example, in Figure 15, RSU_3 knows blind value K_{11} and the blind value of his brothers K_{10}, $K_{4,}$ and K_3, and, therefore, can obtain all keys in its predecessor set $K_5, K_{2,}$ and K_1, i.e., the group key (KG_i). This approach preserves the security of the group key.

Finally, holding parameters $\{\bar{p}, \bar{q}, d\}$ secret, the CCS publishes $\{H, p, \beta, Y_{pub}, Q_{ccs}, T, KG_i, K_{group}\}$, where $K_{group} = K_1, K_2, K_3, K_4 \ldots . K_\tau$.

2nd phase: **EV registration**

When a user decides to recharge their EV, they will use a CWD-WPT station the first step is to register through a secure channel in CCS. The user must select a private key $x_{ev} \in Z_p^*$ and calculate public key $Y_{ev} = T_{x_{ev}}(\beta) mod(p)$. The user and EV data (vehicle charging parameters (VCP) (e.g., battery type, charging level, among others), identity (ID_{EV}), and public key) are sent to the CCS for storage. Finally, a certificate $Cert_{ev} = T_{x_{ccs}}(Q_{ev}) mod(p)$, where $Q_{ev} = T_{H(ID_{ev})}(\beta) mod(p)$, is created by the CCS and sent jointly with Q_{ev} to the EV.

3rd phase: **Tickets Purchase**

A secure channel is used for message exchange. We have assumed each ticket guarantees a specific amount of energy is induced to the EV through the pads. The customer buys several tickets offline with money from their bank account associated with the CCS. Each user can buy multiple charging tickets at once.

Initially, the EV requests the purchase of j tickets from the CCS by sending it the following message:

$$m_1 = \{j, ID_{ev}, Cert_{ev}\}$$

The CCS creates j pairs of random values $\{\varepsilon_1, \varepsilon_2, \dots, \varepsilon_j\} \epsilon\, Z_n^*$ and $\{r_1, r_2, \dots, r_j\} \epsilon\, Z_n^*$; each r_i for $1 \leq i \leq j$ must satisfy property $gcd(r_i, n) = 1$, as in [48]. For each r_i, $\hat{t}_i = T_{r_i}(\beta) mod(p)$, the CCS calculates $gcd(\hat{t}_i, n)$ and checks if $gcd(\hat{t}_i, n) = 1$. If it is not valid, the CCS selects other values. If the validation is correct, the EV obtains $\alpha_i = T_{\varepsilon_i}(\beta) mod(p)$ for each ε_i, and a message m_2 composed of $\Omega = \{\hat{t}_1, \hat{t}_2, \dots., \hat{t}_j\}$ and $A = \{\alpha_1, \alpha_2, \dots., \alpha_j\}$ is sent to the EV:

$$m_2 = \{\Omega, A\}$$

The EV then creates j random values $\{c_1, c_2, \dots, c_j\} \epsilon\, Z_n^*$ and j random tuples $\{(u_1, v_1), (u_2, v_2), \dots., (u_j, v_j)\}$, where u and $v\, \epsilon\, Z_n^*$, and calculates value $\theta_i = c_i \alpha_i$, for all $1 \leq i \leq j$ and the following functions:

$$t_i = T_{u_i + v_i}(\hat{t}_i) mod(p), \text{ for all } 1 \leq i \leq j \quad (57)$$

$$\mu_i = u_i^{-1} \theta_i \hat{t}_i t_i^{-1}, \text{ for all } 1 \leq i \leq j \quad (58)$$

The EV sends message $m_3 = \{U, C\}$ with $U = \{\mu_1, \mu_2, \dots., \mu_j\}$ and $C = \{c_1, c_2, \dots, c_j\}$ to the CCS, which calculates

$$\hat{b}_i = (\mu_i x_{ccs} c_i r_i^{-1} + \hat{t}_i) mod(n), \text{ for all } 1 \leq i \leq j \quad (59)$$

and sends message $m_4 = \{\hat{b}_1, \hat{b}_2, \dots., \hat{b}_j\} = \{\hat{B}\}$ to the EV.

The EV receives m_4 containing $\hat{B}$ and calculates:

$$b_i = \hat{b}_i^{-e} (\hat{b}_i t_i \hat{t}_i^{-1} u_i + v_i t_i) mod(n), \text{ for all } 1 \leq i \leq j \quad (60)$$

Message m_5, containing $B = \{b_1, b_2, \dots., b_j\}$, is then sent by the EV to the CCS:

$$m_5 = \{B\}.$$

CCS then calculates

$$\hat{l_\iota} = (r_i b_i)^d mod(n), \text{ for all } 1 \leq i \leq j \quad (61)$$

and sends message m_6 to the EV:

$$m_6 = \{\hat{L}\}, \text{ where } \hat{L} = \{\hat{l}_1, \hat{l}_2, \dots., \hat{l}_j\}.$$

The EV calculates:

$$o_i = (\hat{l}_i \hat{b}_i) mod(n) \tag{62}$$

Finally, the valid ticket, composed of $(\theta_i,\ t_i, o_i)$, is obtained. Figure 16 summarizes the ticket purchase phase.

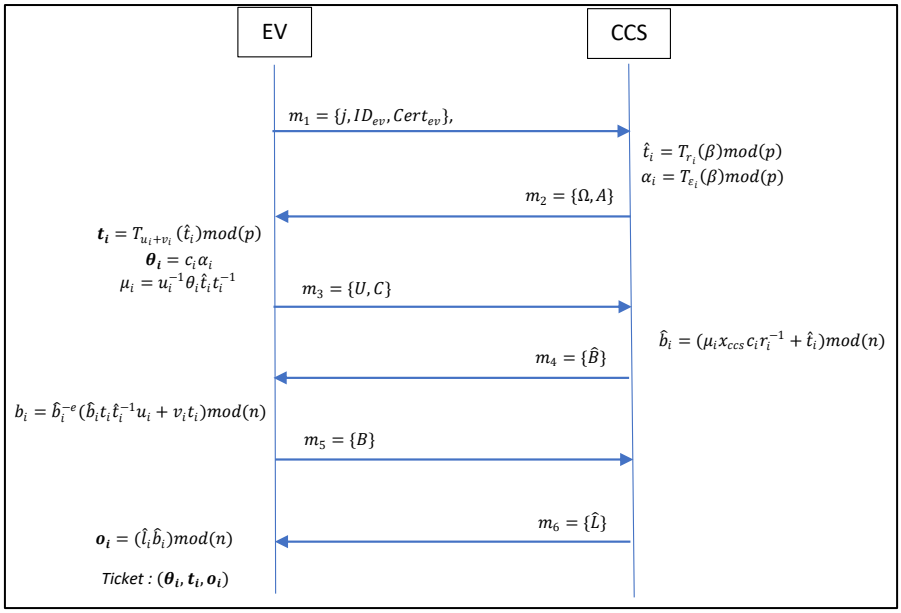


FIGURE 16. TICKET PURCHASE OF THE PROPOSED PROTOCOL PROT_2

4th phase: **Charging Request and Authentication**

This phase describes the process of authentication, verification, and creation of session keys between the CWD-WPT charging station and the EV.

i) **Access to the charging station**

When an EV owner wants to recharge the vehicle at a CWD-WPT charging station and has a valid ticket (θ, t, o), the EV chooses a random number $\sigma_{ev} \in Z_n^*$, calculates $\gamma_{ev} = T_{\sigma_{ev}}(\beta) mod(p)$, and sends an m_1 message to the RSU_1

$m_1 = \{\gamma_{ev}, ts_1, H(\gamma_{ev} || ts_1)\}$, where ts_1 is a timestamp.

When RSU_1 receives message m_1, it checks the timestamp and hash. If the match is valid, a random value $\sigma_{rsu_l} \in Z_n^*$ is chosen and session key $k_{rsu_1-ev} = T_{\sigma_{rsu_1}}(\gamma_{ev}) mod(p)$ is calculated.

On the other hand, RSU_1 calculates the following values so that the EV can obtain the session key and authenticate it:

$$\gamma_{rsu_1} = T_{\sigma_{rsu_1}}(\beta) mod(p), \text{ and} \tag{63}$$
$$\eta_{rsu_1} = T_{x_{rsu_1}}(\varpi) mod(p). \tag{64}$$

where $\varpi = T_{H(\gamma_{rsu_1}, VK, ts_1, ts_2)}(\beta) mod(p)$.

RSU_1 immediately sends message m_2 to the EV.

$$m_2 = \{\gamma_{rsu_1}, VK, ts_2, \eta_{rsu_1}\}$$

After the EV receives m_2, it recalculates the RSU_1*'s* signature with the values received in the message: $\eta_{rsu_1} = T_{H(\gamma_{rsu_1}, ts_1, ts_2)}(Y_{rsu_1}) mod(p)$ and compares with the signature that arrived in message m_2. If the verification is successful, the EV accepts the message sent by the RSU and uses its values to obtain session key $k_{rsu_1-EV} = T_{\sigma_{ev}}(\gamma_{rsu_1}) mod(p)$ and the session key to encrypt. It then sends RSU_1 message m_3 containing ticket (θ, t, o) and a timestamp.

$$m_3 = \{\theta, t, o, ts_3\}_{k_{rsu_1-ev}}$$

RSU_1 deciphers the message and forwards the ticket to the fog server through a message m_4, which is deciphered with session key k_{fs-rsu_1}. The timestamp is checked and the ticket (θ, t, o) validity is immediately verified:

$$\left[T_{o^e mod(n)}(\beta)\right]^2 + \left[T_{\theta mod(n)}(Y_{pub})\right]^2 + [T_t(t)]^2 = \left(2T_{o^e}(\beta).T_\theta(Y_{pub}).T_t(t) + 1\right) mod(p). \quad (65)$$

If the ticket is validated, the FS adds a leaf to the binary tree where it has located the client EV (see Figure 17).

Managing the binary tree RSUs group ensures security through the update of the group key when a new EV is removed or added to the group. Towards updating the group key after an EV has been added or removed from the tree, all members individually calculate the new blind keys along the affected route of the binary tree. In what follows is the description of this operation [79].

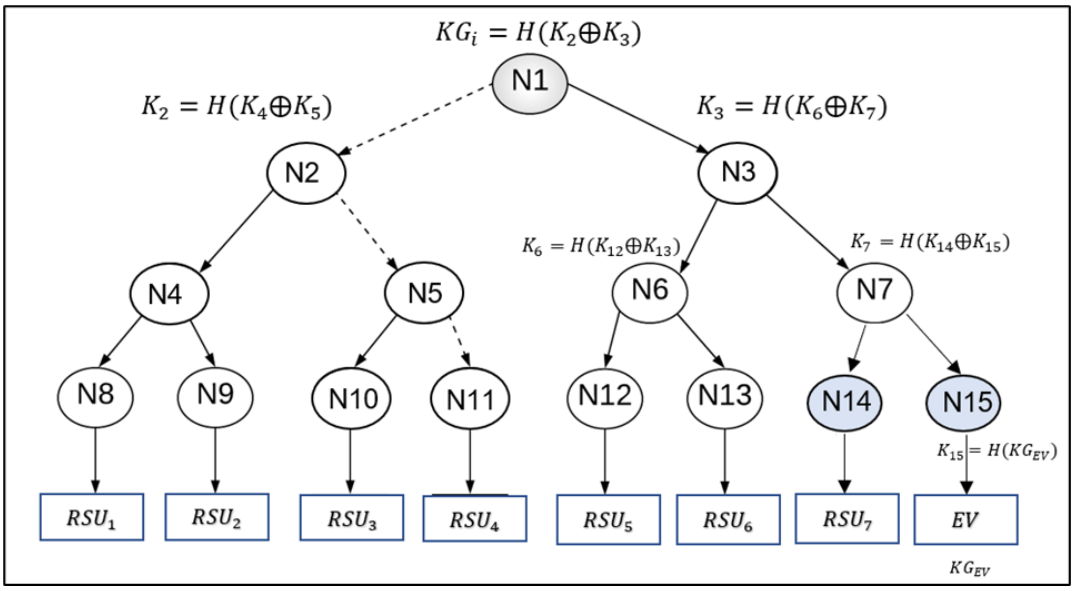


FIGURE 17. BINARY TREE WITH EV CLIENT

Whenever a new EV joins the group, it is associated with the leaf node of a binary tree. On the other hand, when a leaf node becomes the parent of two leaf nodes (right and left), the element (RSU) associated with the new parent node is associated with the left leaf node and the EV is associated with the right leaf node. A new group key is then generated. Below is an example of the addition of an EV to the Tree.

• leaf node N7 becomes a parent node and creates two leaf nodes (N14 and N15). RSU_7, which was associated with N7, is now associated with N14 (right leaf node) and EV_i is associated with leaf node N15 (right leaf node) (see Figure 17). When N7 has a new key, the binary tree must recalculate the group key.

The FS sends verification key KG_{i-ev} and part of ticket θ to the RSUs of the charging station via an m_5 encrypted message with the group key.

$$m_5 = \{ KG_{i-ev}, \theta \}_{K_{G-rsu}}$$

Additionally, the FS sends the tree information and EV group key to RSU_1, which groups such information, adds a random seed λ_1 and the number of pads ψ_1 it controls, and sends message m_6 to the EV.

$$m_6 = \{ KG_{i-ev}, \lambda_1, \psi_1, ts_4 \}_{k_{rsu-ev}}$$

Simultaneously, RSU_1 sends message m_7 encrypted (with the group key k_{G-pad}) to all pads through a broadcast. m_7 contains public hash chain verication key $k_{PH} = H^{\psi+1}(\lambda)$, which is used to verify and authenticate to the EV [35].

$$m_7 = \{k_{PH}, ts_4\}_{K_{G-pad}}$$

Hash chain $H^{\psi}(\lambda)$ is then computed using ψ and λ values. In an RSU, each group of pads decrypts message m_7 with the group key, obtaining $k_{PH} = H^{\psi+1}(\lambda)$ (public hash chain verification key). A hash chain H^{ξ} (for $0 \leq \xi \leq \psi + 1$), containing a key, is sent by the electric vehicle to one of the pads through message $m_8 = \{H^{\xi}\}$; the pad applies hash function ($H^z(H^{\xi})$) z times, with $z = \xi - \psi + 1$, to verify the key validity.

Value $H^z(H^{\xi})$ is compared with the verification key (public key hash string). If the check is valid, the pad checks the status of the key in the revocation list. If the key is not in the list, the pad accepts the key from the EV and revokes it to avoid its reuse. Figure 18 shows the charging in the first RSU.

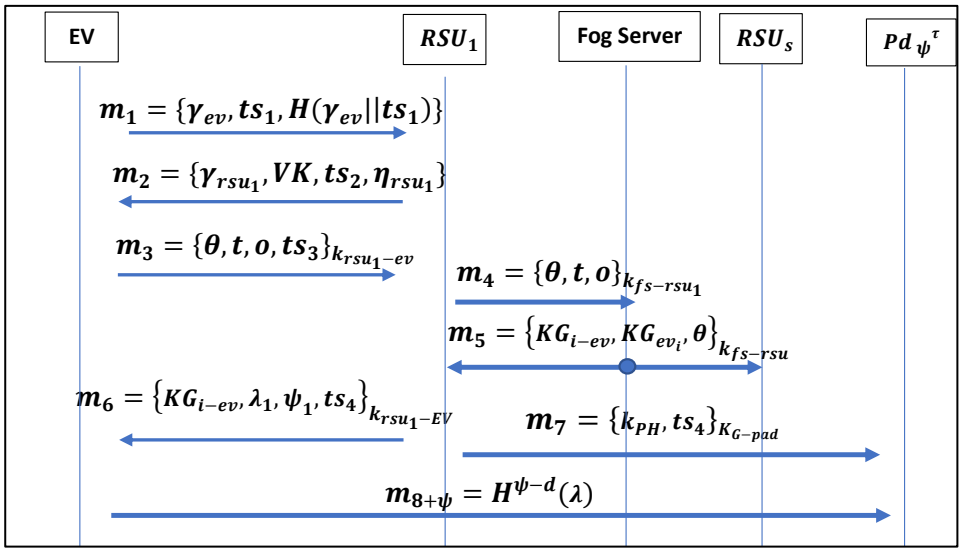


FIGURE 18. CHARGING IN THE FIRST RSU OF THE PROPOSED PROTOCOL PROT_2

ii) Authentication of an RSUs charging station

The authentication of a second RSU is explained in what follows towards a simpler description of the protocol. For other RSUs and associated pads, the authentication process with the EV is similar to the one described below.

Towards authenticating with RSU_2 , the EV sends message $m_{9+\psi_1}$ containing θ, which is the ID value of the ticket, and random value q encrypted with the group key

$$m_{9+\psi_1} = \{\theta, \{q, ts_5\}_{KG_{i-ev}}\}$$

RSU_2 deciphers the message with the key associated with the Ticket ID value θ and checks the timestamp. Finally, it sends message $m_{10+\psi_1}$ encrypted back to the EV and, simultaneously, message $m_{11+\psi_1}$to its pad group.

$$m_{10+\psi_1} = \{\lambda_2, \psi_2, HMAC_{EV}^1, ts_6\}_{KG_{i-ev}},$$

where $HMAC_{EV}^1 = H(q, \lambda_2, \psi_2, ts_5, ts_6\ )$

$$m_{11+\psi_1} = \{k_{PH}, ts_7\}_{K_{G-pad}}$$

EV decrypts message $m_{10+\psi_1}$ and calculates $HMAC_{EV}^1{}'$ to verify and authenticate RSU_2 . Simultaneously, RSU_2 sends message $m_{11+\psi_1}$ with the hash chain check key (K_{PH}) to all the pads it manages.

After receiving $m_{10+\psi_1}$, the EV performs the same process applied to m_6 and send $m_{12+\psi_1}$ to each pad. Figure 19 illustrates the authentication process in the other RSUs.

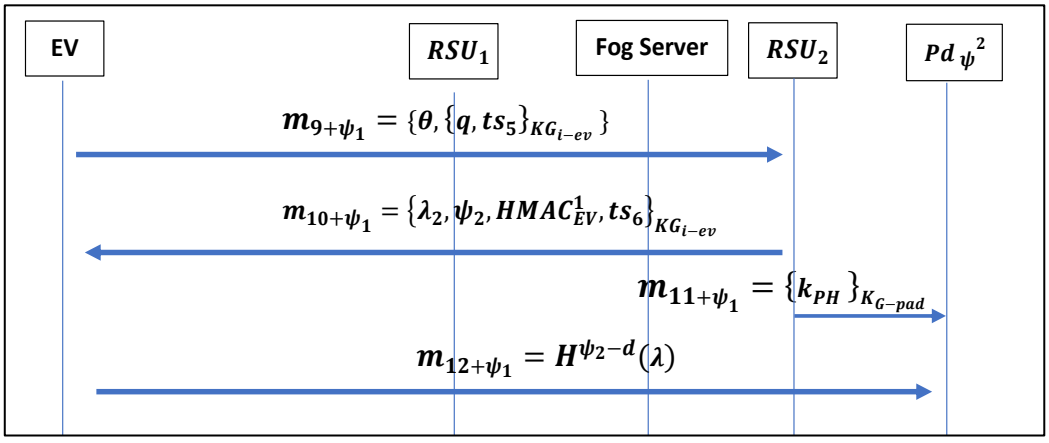


FIGURE 19. CHARGING PROCESS IN THE OTHER RSUS OF THE PROPOSED PROTOCOL PROT_2

The protocol contains a generalization of the scheme designed by Tahat et al. [24] in terms of some mathematical operations. The following changes have been made:

- Blind signatures are made on multiple tickets sent in a same message;
- Element "c" previously shared in Tahat et al. [48] is a random value in our protocol shared during the signing of the ticket; additionally, it is multiplied by "α" for the creation of blinding value "θ".

Our protocol uses blinding factor "θ" in the ticket validation with no pre shared element between entities, as in Tahat et al. [48], thus guaranteeing the confidentiality and privacy of both user and EV.

In what follows are the mathematical proofs of the pseudo blind pseudonym (Ticket validation) and the verification of the signature of the first RSU.

$$\begin{aligned}\left[T_{o^e mod(n)}(\beta)\right]^2 + \left[T_{\theta mod(n)}\left(Y_{pub}\right)\right]^2 + [T_t(t)]^2 \\ = \left(2T_{o^e}(\beta).T_\theta\left(Y_{pub}\right).T_t(t)+1\right)mod(p),\end{aligned} \tag{66}$$

Since

$$o^e\left(mod(n)\right) = (\hat{l}_i\hat{b}_i)^e, \tag{67}$$
$$= \left(r_i^d b_i^d \hat{b}_i\right)^e, \tag{68}$$
$$= r_i b_i \hat{b}_i^e, \tag{69}$$
$$= r_i \hat{b}_i^{-e}\left(\hat{b}_i t_i \hat{t}_i^{-1} u_i + v_i t_i\right)\hat{b}_i^e, \tag{70}$$
$$= r_i\left(\hat{b}_i t_i \hat{t}_i^{-1} u_i + v_i t_i\right), \tag{71}$$
$$= r_i(\mu_i x_{ccs} c_i r_i^{-1} + \hat{t}_i)\, t_i \hat{t}_i^{-1} u_i + v_i t_i), \tag{72}$$
$$= (\mu_i x_{ccs} c_i + \hat{t}_i r_i)\, t_i \hat{t}_i^{-1} u_i + r_i v_i t_i), \tag{73}$$
$$= ((u_i^{-1}\theta_i \hat{t}_i t_i^{-1}) x_{ccs} c_i + \hat{t}_i r_i) t_i \hat{t}_i^{-1} u_i + r_i v_i t_i), \tag{74}$$
$$= (\theta_i x_{ccs} c_i + u_i t_i r_i + r_i v_i t_i) mod(n). \tag{75}$$

then:

$$\left[T_{o_i^e mod(n)}(\beta)\right]^2 + \left[T_{\theta_i mod(n)}\left(Y_{pub}\right)\right]^2 + \left[T_{t_i}(t)\right]^2 \tag{76}$$
$$= \left[T_{(\theta_i x_{ccs} c_i + u_i t_i r_i + r_i v_i t_i) mod(n)}(\beta)\right]^2 + \left[T_{\theta_i mod(n)}\left(T_{x_{ccs}}(\beta)\right)\right]^2 + \left[T_{t_i}\left(T_{u_i+v_i}\left(\hat{t}_i\right)\right)\right]^2, \tag{77}$$
$$= \left[T_{(\theta_i s_{ccs} c_i + u_i t_i r_i + r_i v_i t_i) mod(n)}(\beta)\right]^2 + \left[T_{\theta_i mod(n)}\left(T_{x_{ccs}}(\beta)\right)\right]^2 + \left[T_{t_i}\left(T_{u_i+v_i}\left(T_{r_i}(\beta)\right)\right)\right]^2, \tag{78}$$
$$= \left[T_{(\theta_i s_{ccs} c_i + u_i t_i r_i + r_i v_i t_i)}(\beta)\right]^2 + \left[T_{\theta_i} T_{x_{ccs}}(\beta)\right]^2 + \left[T_{t_i} T_{u_i+v_i}\, T_{r_i}(\beta)\right]^2, \tag{79}$$
$$= \left[T_{(\theta_i\, s_{ccs}\, c_i + u_i\, t_i\, r_i + r_i\, v_i\, t_i)}(\beta)\right]^2 + \left[T_{\theta_i x_{ccs}}(\beta)\right]^2 + \left[T_{t_i(u_i+v_i) r_i}(\beta)\right]^2, \tag{80}$$
$$= \left[T_{(\theta_i\, s_{ccs}\, c_i + u_i\, t_i\, r_i + r_i\, v_i\, t_i)}(\beta)\right]^2 + \left[T_{\theta_i x_{ccs}}(\beta)\right]^2 + \left[T_{t_i u_i r_i + t_i v_i r_i}(\beta)\right]^2. \tag{81}$$

Let us consider $a = \theta_i\, s_{ccs}\, c_i + u_i\, t_i\, r_i + r_i\, v_i\, t_i$; $b = \theta_i x_{ccs}$; and $c = t_i u_i r_i + t_i v_i r_i$.

According to Theorem 3, if $a = b + c$, the following is valid:

$$\left[T_{(\theta_i\, s_{ccs}\, c_i + u_i\, t_i\, r_i + r_i\, v_i\, t_i)}(\beta)\right]^2 + \left[T_{\theta_i x_{ccs}}(\beta)\right]^2 + \left[T_{t_i u_i r_i + t_i v_i r_i}(\beta)\right]^2 \tag{82}$$
$$= \left(2T_{(\theta_i\, s_{ccs}\, c_i + u_i\, t_i\, r_i + r_i\, v_i\, t_i)}(\beta).T_{\theta_i x_{ccs}}(\beta).T_{t_i u_i r_i + t_i v_i r_i}(\beta)+1\right)mod(p), \tag{83}$$
$$= \left(\left[2T_{o_i^e}(\beta)\right].\left[T_{\theta_i} T_{x_{ccs}}(\beta\right].\left[T_{t_i} T_{u_i+v_i}\, T_{r_i}(\beta)\right]+1\right)mod(p), \tag{84}$$
$$= \left(\left[2T_{o_i^e}(\beta)\right].\left[T_{\theta_i}\left(T_{x_{ccs}}(\beta)\right)\right].\left[T_{t_i}\left(T_{u_i+v_i}\left(T_{r_i}(\beta)\right)\right)\right]+1\right)mod(p), \tag{85}$$
$$= \left(2\left[T_{o_i^e}(\beta)\right].\left[T_{\theta_i}(Y_{pub})\right].\left[T_{t_i}\left(T_{u_i+v_i}\left(\hat{t}_i\right)\right)\right]+1\right)mod(p), \tag{86}$$
$$= \left(2T_{o_i^e}(\beta).T_{\theta_i}\left(Y_{pub}\right).T_{t_i}(t_i)+1\right)mod(p). \tag{87}$$

According to Theorem 2.3 of [48] , if $a = b + c$, the following is valid:

$$\left[T_{(\theta_i\, s_{ccs}\, c_i + u_i\, t_i\, r_i + r_i\, v_i\, t_i)}(\beta)\right]^2 + \left[T_{\theta_i x_{ccs}}(\beta)\right]^2 + \left[T_{t_i u_i r_i + t_i v_i r_i}(\beta)\right]^2, \tag{88}$$

$$= \left(\left[2T_{o_i^e}(\beta)\right].\left[T_{\theta_i} T_{x_{ccs}}(\beta)\right].\left[T_{t_i} T_{u_i + v_i}\, T_{r_i}(\beta)\right] + 1\right) mod(p), \tag{89}$$

$$= \left(2T_{o_i^e}(\beta).\, T_{\theta_i}(Y_{pub}).\, T_{t_i}(t_i) + 1\right) mod(p). \tag{90}$$

4.4.1. Comparative Performance Evaluation

A performance analysis conducted involved communication, computational, and energy costs of the proposed protocol, and independence of the authentication processes among its different entities was considered through the application of such processes in different places and time periods. For example, an EV can authenticate with the RSU several meters away; however, to authenticate with the pads, it must be a few centimeters away from them.

The analysis considering the evaluation of three types of costs is described in the sequence.

4.4.1.1. Communication Costs

The communication cost calculation considers the bytes of the message transmitted by the network during the authentication process, but not the headers or control bits inherent to the communication protocol used. The number of bytes of each message and the number of messages are taken into account.

TABLE 9. SYMBOLS AND COSTS IN BYTES (PROPOSED PROTOCOL PROT_2)

Symbol	Description	Length (Bytes)
ID	Identification	128
PID	Pseudo Identity	32
$H()$	Hash function	32
x, S	Private key	32
Y, Q	Public key	32
k	Session key	32
η	Digital signature	32
(θ, t, o)	Ticket	96
τ	Number of RSUs for fog server	8
ψ	Number of pads for RSU	8
λ	Seed	20
ts	Timestamp	8
VK	Verification key	32
$p, n, e, d, \mathrm{c}, \gamma, q$	Prime numbers	32
HMAC	Hash-based message authentication code	32

Table 9 shows the values in bits of the variables used in the protocol (values taken from Rabieh and Wei [25]).

According to Table 4, $\boldsymbol{\tau}$ RSUs, $\boldsymbol{\psi}$ pads (for each RSU), and $\boldsymbol{n}$ EVs are considered for the calculation of the communication costs. A comparison of the costs among our protocol and those of Pazos-Revilla et al.[58] and Li et al. [60] is shown in Table 9.

TABLE 10. COMPARISON OF COMMUNICATION COSTS IN BYTES (PROPOSED PROTOCOL PROT_2)

Message	Pazos-Revilla et al. [58]	Li et al. [60]	Proposed Protocol PROT_2
M1	$32n$	$140n$	$74n$
M2	$128n$	$72(n * \psi)$	$74n$
M3	$128n$	$136n$	$104n$
M4	$96n$	$240n$	$96n$
M5	$32n$	$64(n * \psi)$	$64n$
M6	$40n$	$32(n * \psi)$	$60n$
M7	40τ	$264n$	$40n$
M8	$32(n * \tau)$	$32(n * \psi)$	$32(n * \psi)$
M9	$32(n * \tau)$	$176n$	$74(n * \tau - 1)$
M10	$32(n * \tau * \psi)$	$--$	$68(n * \tau - 1)$
M11	$--$	$--$	$32(n * \tau - 1)$
M12	$--$	$--$	$32(n * \tau - 1 * \psi)$
Total	$\boldsymbol{n(456 + \tau(64 + 32\psi)) + 40\tau}$	$\boldsymbol{n(956 + 200\psi)}$	$\boldsymbol{n(512 + 32\psi + (\tau - 1) * (174 + 32\psi))}$

In this sub-section, we consider a CWD-WPT charging system with the same characteristics of the system described in sub-section 4.3.1.1 for the comparison of our protocol. with those of Pazos-Revilla et al.[58] and Li et al. [60]. Figure 20 shows the communication cost of our protocol is better than that of Li el al. [60] and very similar to that of Pazos-Revilla et al.[58]. However, depending on the values of *n (*EVs*) ,τ(* 7 RSUs*)* and *ψ(*750/RSU*)*, small differences may occur, thus affecting the structure of the CWD-WPT charging station.

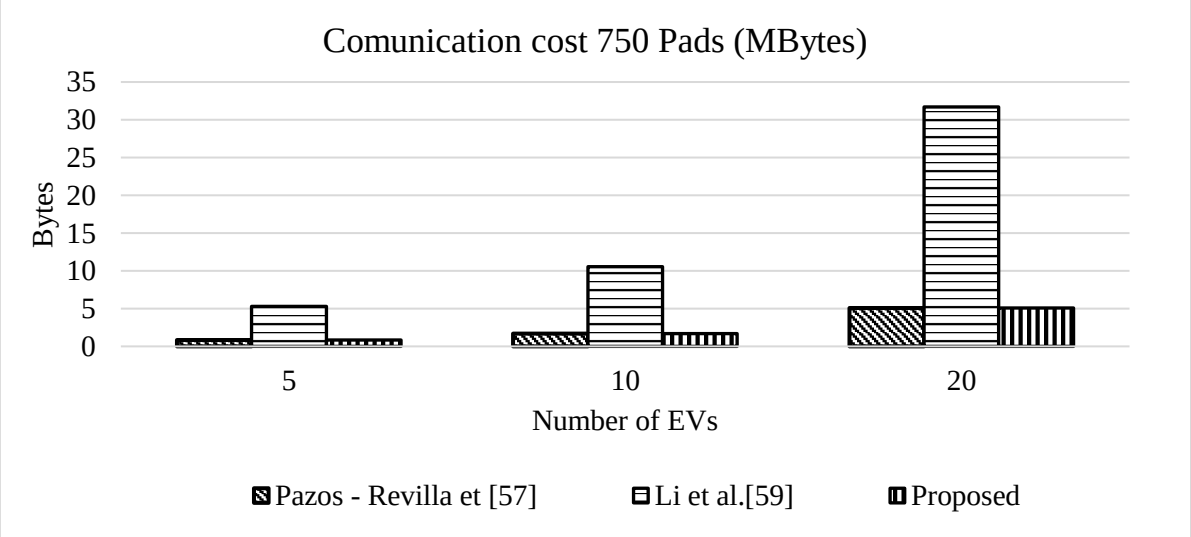


FIGURE 20. COMPARISON OF COMMUNICATION COSTS (PROPOSED PROTOCOL PROT_2)

4.4.1.2. Computational Costs

The computational costs are evaluated taking into account the time necessary for carrying out the unitary operations, which are estimated according to the processing power of each entity. Such cost values are based on experiments made on common computing platforms and adopted for comparing the performance of authentication protocols. In this sense, towards defining a reference architecture for the evaluation of authentication protocols, Tao et al. [52] obtained the computational cost of each unitary operation taking into account 3 hardware types:

- mobile equipment (processor Qualcomm (R) Octa core 1.5 GHz, 2G RAM).
- a Desktop (Intel (R) Dual core processor 3.1 GHz, 4GB RAM), and
- a Server (Intel (R) Hexa core processor 1.6 GHz and 16G RAM).

For our study, such hardware types correspond to EV/pad, RSUs, and FS/CSP, respectively.

The methodology adopted considers each unitary operation requires a specific computational effort, whose time cost is multiplied by the number of times it is performed, as required for the performance evaluation of different authentication protocols. Table 11 shows the execution times of the cryptographic unitary operations used by the different protocols, according to the values provided in [18], [45], and [46].

TABLE 11. COSTS IN ***MS*** OF EACH OPERATION AND ENTITY CONSIDERED (PROPOSED PROTOCOL PROT_2)

Entity	Costs (ms)						
	Tmp_{mul}	Tmp_{exp}	Tmp_{pair}	Tmp_{hash}	Tmp_{g-ptns}	Tmp_{v-ptns}	Tmp_{Chaos}
EV/Pad	0.29	0.5	0.75	$0.3x10^3$	0.03	0.021	0.95
RSUs	0.22	0.38	0.57	$0.2x10^3$	0.011	0.015	0.07
FS/ CSP	0.17	0.31	0.46	$0.1x10^3$	0.009	0.01	0.05

Table 12 shows a comparison of the number of operations performed by the three protocols. Similarly to the protocol of Pazos-Revilla et al. [58], our scheme performs the most processing work in the entity with improved computational characteristics (the FS, in our case). On the other hand, EV, RSU, and the pads have fewer computing capacities, and, therefore, conduct less complete operations, which helps reduce the latency of the system. The processing of the protocol of Li et al. [60] is concentrated on the EV and pads, requiring a higher computational cost compared to our protocol.

TABLE 12.COMPARISON OF COMPUTATIONAL COSTS (PROPOSED PROTOCOL PROT_2)

Protocols	**EV**	**CSP BNK/ /FS**	**RSU/ Pad Owner**	**Pad**
Pazos-Revilla et al. [58]	$5Tmp_{exp}$ + $(3+2\psi)Tmp_{hash}$ +$2Tmp_{pair}$	$(8n+3)Tmp_{exp}$ +$(4n+3)Tmp_{mul}$ +$2nTmp_{pair}$	$2Tmp_{pair}$ + $3nTmp_{hash}$	$2n(\psi)Tmp_{hash}$
Li et al. [60]	$(1+\psi)Tmp_{g-ptns}$ + $2Tmp_{v-ptns}$	$(2n+1)Tmp_{g-ptns}$	$(n)Tmp_{v-ptns}$	$n(\psi)Tmp_{v-ptns}$
PROT_2	$3Tmp_{chaos} + (1+\psi)Tmp_{hash}$	$4nT_{exp} + 6n\,Tmp_{chaos}$ +$2n\,Tmp_{mul}$	$2nT_{hash}$ +$4n\,Tmp_{chaos}$	$n(\psi)T_{hash}$

Figure 21 depicts a comparison of computational costs of the authentication phase among our protocol and those of Pazos-Revilla et al. [58] and Li et al. [60]. The cost of our protocol is better and the use of chaos-based cryptography shows a better computational cost compared to schemes such as bilinear pairing encryption.

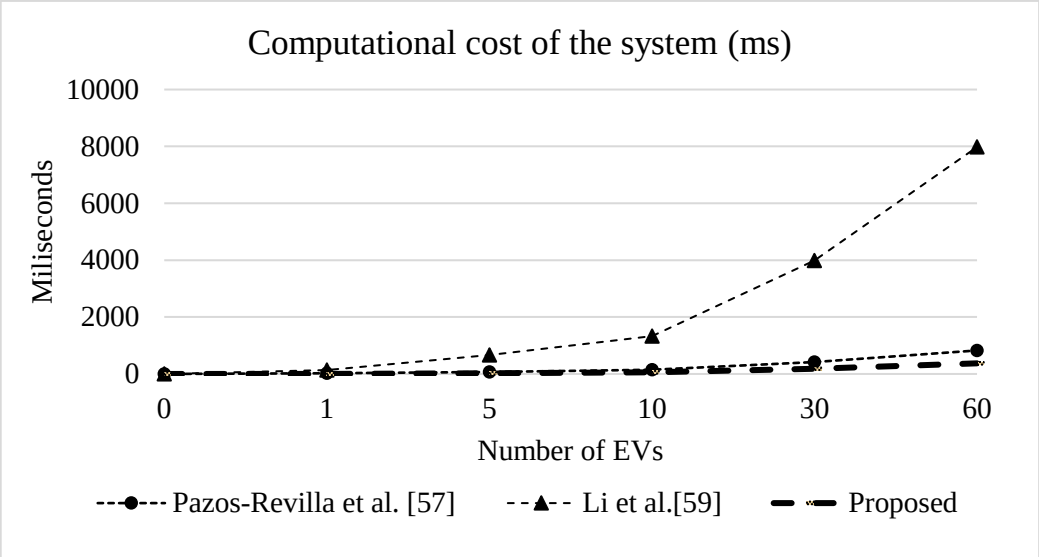


FIGURE 21 COMPUTATIONAL COSTS (PROPOSED PROTOCOL PROT_2)

4.4.1.3. Energy Costs

The costs of the energy consumed in the execution of cryptographic operations in the protocols were compared. Equation $\boldsymbol{E_C} = T_{EX} * W$ (joules units),where T_{EX} is the execution time in ms and W is the maximum power CPU, calculated the energy costs. W = 10.88 watts [76][77] was assumed for the comparison of the energy costs of the proposed protocol with those of [58] and [60] (see Table 13). According to Figure 22, our protocol consumed the lowest energy.

TABLE 13. COMPARISON OF ENERGY COSTS (PROPOSED PROTOCOL PROT_2)

Protocols	Equation
Pazos-Revilla et al. [58]	$E_{cost} = \left((\mathbf{13}\boldsymbol{n} + \mathbf{3})\boldsymbol{Tmp_{exp}} + (\mathbf{6}+\mathbf{4}\boldsymbol{\psi})\boldsymbol{nTmp_{hash}} + (\mathbf{2} + \mathbf{2}\boldsymbol{n})\boldsymbol{Tmp_{pair}} + (\mathbf{4}\boldsymbol{n} + \mathbf{3})\boldsymbol{Tmp_{mul}}\right) * 10{,}88W$
Li et al. [60]	$E_{cost} = \left(((\mathbf{3} + \boldsymbol{\psi})\boldsymbol{n} + \mathbf{1})\boldsymbol{Tmp_{g-ptns}} + (\mathbf{3} + \boldsymbol{\psi})\boldsymbol{nTmp_{v-ptns}}\right) * 10{,}88W$
PROT_2	$E_{cost} = \left(\mathbf{13}\boldsymbol{Tmp_{chaos}} + (\mathbf{3} + \mathbf{2}\boldsymbol{\psi})\boldsymbol{nTmp_{hash}} + \mathbf{2}\boldsymbol{nTmp_{mul}} + \mathbf{4}\boldsymbol{nT_{exp}}\right) * 10{,}88W$

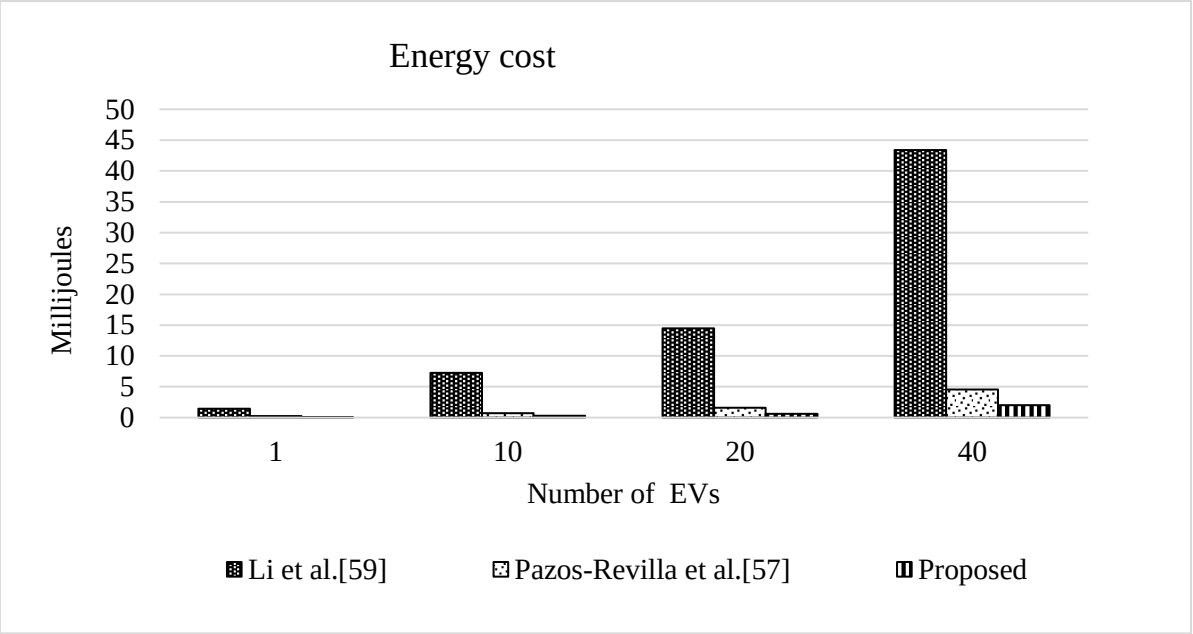


FIGURE 22. ENERGY COSTS COMPARISON (PROPOSED PROTOCOL PROT_2)

4.5. Security Verification of the Proposed Protocols PROT_1 and PROT_2

This section reports an analysis and a comparison of the Proposed Protocols PROT_1 and PROT_2 with other authentication protocols of a CWD-WPT system regarding performance and security characteristics. The security analysis is based on security properties and possible attacks, whereas the performance analysis is based on the evaluation of communication and computational costs.

4.5.1. Discussion about Security Properties

Below is an analytical description of the security attributes, like mutual authentication, privacy preservation and integrity protection guaranteed by our protocols PROT_1 and PROT_2 and a description of the way they resist attacks.

1) Privacy preservation: during the ticket purchase process, in both Protocols 1 and 2, the CCS keeps the identity of the purchasing user confidential. When the user uses the charging station, FS, RSUs, and pads cannot obtain the user's identity from the ticket.

2) Mutual Authentication: this process is established among FS, RSU and EVs.

In the case of protocol nº 1, the EVs authenticate FS by verifying message (m_2) signature. FS authenticates the valid ticket of an EV by verifying the blind signature sent in message 3 and using public parameters of the system. The RSU authenticates the EV by calculating the hash of message 6 containing an $\boldsymbol{\alpha_1}$ (delivered by the FS to the EV in message 4, and the RSU in message 6) sent by the EV. EVs authenticate to RSUs by verifying message 7 HMAC.

In the case of protocol nº 2, the FS and RSU_1 authenticate to the valid EV through the ticket sent in message m_4, and the EV authenticates to FS and RSU_1 through η_{rsu_1} signature. The EV authenticates

the other RSUs using KG_{i-ev} key and theta factor, and the other RSUs authenticate the EV through the MAC containing the $\boldsymbol{q}$ element and ts_5, sent in $m_{10+\psi_1}$.

3) Protection to integrity: in both Proposed Protocols PROT_1 and PROT_2, integrity is guaranteed by means of hash function and digital signatures. The system can identify whether an adversary manipulates the message by verifying the hash function value or the digital signature of the message.

4) Perfect Forward Secrecy (PFS): the proposed protocols guarantee PFS as follows:

a. For Proposed Protocol PROT_1

- In the process of creating session key $k_{(fs-EV)}$ between EV and FS to encrypt the messages, the random elements φ_{EV}, φ_{fs} and a blind message signature are used. Even if the session key $k_{(fs-EV)}$ is compromised, the previous messages cannot be recovered because of the CDH problem;
- In the process of creating a session key $k_{(RSU-EV)}$ between the EV and the RSU to encrypt the messages, the random elements α_1, α_2 and $PID2_1$ are used. Even if some or all of the random values are committed and the attacker manages to recreate the session key $k_{(RSU-EV)}$, previous messages cannot be recovered due to the CDH problem;
- In the process of creating the key $H^{\psi(v)}$ between the EV and the pads, in the worst case when the seed ν is compromised, the attacker will not be able to decipher the previous messages;
- If the CCS (X_{ccs}) private key is compromised, an attacker will not be able to recreate previous session keys and therefore decrypt old messages due to the random values used for generating session keys.

b. For Proposed Protocol PROT_2

- random elements, such as γ_{ev} and γ_{rsu} are used for the creation of the session key between EV and RSU (k_{rsu_1-ev}). Even if session key k_{rsu-ev} is compromised, the recovery of previous messages is very difficult, due to the CDH problem;
- if the λ seed created for the authentication of EVs and pads is compromised, the attacker will not be able to decipher previous messages; and
- if the CCS (S_{ccs}) private key is compromised, an attacker cannot recreate previous session keys and, therefore, decrypt old messages due to the random values used for the generation of session keys.

5) Unlinkability: in both Protocols 1 and 2, the ticket cannot be linked with a certain EV, since the ticket is blindly signed by CCS and verification by FS is performed with a system of public values.

6) Double Spending:

In the case of protocol PROT_1, the double spending is avoided when an EV uses PID_{ev} and its signatures (C ', S') to authenticate to the fog server, PID_{ev} is revoked and published on a fog server's revocation list. In the authentication process, the fog server checks if PID_{ev} is on the list for terminating the continuing authentication process at the charging station. The same occurs in the EV authentication process in the RSU. PID_{ev} is revoked and published on a revocation list of RSUs.

On the other hand, in protocol PROT_2 the double spending is avoided when an EV uses ticket $(\theta,\ t, o)$ to authenticate with the fog server, the ticket is invalidated through its addition to the system's revocation list. Throughout the EV authentication process, the ticket is checked by the fog server in the revocation list; if it is in the list, the system ends the authentication process and does not

provide the service to the vehicle. Revocation lists are also used in the RSU and EV authentication process for preventing Double Spending.

4.5.2. Resistance to attacks

Below are the different types of attacks that can affect the VANET network and a description of the way our protocol can resist them:

Impersonation: in both proposed protocols, the charging station can validate the authenticity of a ticket; therefore, the system can detect if an attacker is trying to access it with a false ticket and expels it. On the other hand, the use of random values for ticket generation will prevent an attacker from accessing the system with an old ticket.

MitM: In the case of protocol PROT_1, the use of digital signatures for the verification of the authenticity and integrity of messages m_2 and m_7 ensures that an MitM attack cannot be successful. On the other hand, when the EV performs an authentication process with the RSU, the EV sends a hash chain generated by the seed α_1 in message m_6, taking into that account only an authentic EV can generate the valid hash chain, the MitM attack is mitigated.

On the other hand, the protocol PROT_2 uses HMAC functions and chaos-based signatures to ensure integrity and prevent MitM attacks. In the authentication process between the EV and the RSU, only one EV is valid for generating an HMAC that contains the λ seed, thus avoiding the MitM attack.

Masquerade attack: the Proposed Protocols 1 and 2 are safe against server masking attacks, because an attacker cannot represent the response messages that are sent by the FS or RSU. The FS and RSU sign the contents of the response messages with their private key, so an attacker cannot recreate the signature of the response messages because they do not have the FS or RSU private key.

Replay and Injection: in both proposed protocols 1 and 2, timestamps and random numbers are used in the messages it avoids replay attacks and hash functions and digital signatures can alert about the injection of data in the messages.

Known key: the Proposed Protocols PROT_1 and PROT_2 generate tickets which can be used only once. The ticket is added to the revocation list after its validity has been checked. Both system and EV generates random values for to create session keys, i.e., new session keys are generated for every new ticket for EV communication with the charging station, thus preventing an attacker from charging his/her car using old keys they may know.

DoS: In both Proposed Protocols 1 and 2 DoS attacks can affect the fog server and RSUs. In the first case, the fog server resists DoS attacks by validating tickets with public system parameters and revocation lists.

- In the Proposed Protocol PROT_1, RSUs resist DoS attacks by efficiently validating connection requests using an HMAC code and verifying the auth variable $\boldsymbol{\alpha_1}$ in the revocation lists. Only users previously authenticated by the fog server have a valid $\boldsymbol{\alpha}$ **(alpha)** to create a valid HMAC. If an attacker attempts to connect to the RSU using an already used HMAC or a false HMAC, the RSU rejects the communication.

- In the Proposed Protocol PROT_2, RSUs resist DoS attacks by validating the KG_{i-ev} key as it can only be generated by the ticket owner. Therefore, RSU can detect the attack and close

the connection from an access request from an attacker who is using an old or fake session key or ticket.

Resistance to password guessing attack: in both proposed protocols, whenever an EV wishes to access the system, it uses a ticket generated with random elements, similarly to the session keys used in the exchange of messages.

Random number leakage attack: in both proposed protocols, the following operations and controls in relation to the PRNG system [51] are used to prevent this type of attack:

- A hash function will be executed on the inputs that are counted with a timestamp;
- A hash function will be executed on the PRNG outputs;
- In a period of random time, a new initial PRNG state will be generated;
- Smart seed will be used at the starting points of the PRNG.

Unlinkability: No entity can link PID_{ev} with a single EV, because the CCS blindly signs this value on the ticket "c_i" in the case of Proposed Protocol PROT_1, and "θ" in the case of Proposed Protocol PROT_2. Moreover, the fog server checks the blind signature only with public parameters of the system.

Privileged insider attack: to prevent this type of attack, the company must establish security policies, internal processes and mechanisms for the prevention and detection of attacks. The following is a set of policies to be implemented in the system to prevent such attacks or mitigate damages in the proposed protocols [80]:

- Awareness of security: the company's security policies and procedures must be known to all internal staff and external partners;
- Classification of duties: it is necessary to classify the duties of employees and employers, to prevent or detect the attacks effectively;
- Whirling of duties: when you have several important jobs, you should have several employees with the knowledge of the execution of these jobs; in each time period, these officials have to turn to different jobs to avoid malicious actions;
- Limited privileges: limited access privileges (physical and in systems) must be given to officials to restrict access to confidential information or important company equipment;
- Encrypt sensitive data: confidential data must be encrypted and stored in secure locations. The company must be backed up in the event that the system data is corrupted;
- Defense in depth: a layered security policy must be implemented, where each layer has specific tasks for system protection.

Table 14 shows a comparison of the security analysis among our protocol and other schemes for authentication for CWD-WPT load stations.

Table 14. Comparison of security properties (Proposed Protocols PROT_1 and PROT_2)

Properties and Attacks	[23]	[24]	[25]	[58]	[60]	[57]	[59]	Proposed Protocol PROT_1	Proposed Protocol PROT_2
Mutual authentication and	Y	Y	Y	Y	Y	Y	Y	**Y**	**Y**
key agreement	Y	Y	Y	Y	Y	Y	N	**Y**	**Y**
Confidentiality	Y	Y	Y	Y	Y	N	Y	**Y**	**Y**
Integrity	U	U	U	Y	Y	Y	U	**Y**	**Y**
Privacy	Y	Y	Y	Y	Y	N	Y	**Y**	**Y**
Forward secrecy	U	U	U	U	U	U	U	**Y**	**Y**
Unlinkability	U	U	Y	Y	U	U	Y	**Y**	**Y**
Double spending	U	Y	Y	Y	U	U	Y	**Y**	**Y**
Impersonation attack	Y	U	U	Y	Y	Y	Y	**Y**	**Y**
Man in the middle attack	U	Y	U	Y	U	Y	Y	**Y**	**Y**
Masquerade attack	U	U	U	U	U	U	U	**Y**	**Y**
Replay attack	Y	U	Y	Y	Y	N	Y	**Y**	**Y**
Injection attacks	U	U	U	U	U	U	U	**Y**	**Y**
Know key attack	Y	U	U	U	U	U	U	**Y**	**Y**
DoS attack	U	U	U	U	U	Y	U	**Y**	**Y**
Resistance password guessing attack	U	U	U	Y	U	U	U	**Y**	**Y**
Random number leakage attack	U	U	U	U	U	U	U	**Y**	**Y**
Privileged insider attack	U	U	U	U	U	U	Y	**Y**	**Y**

* Y: Yes; *N: No; *U: Untreated

4.5.3. AVISPA Verification

The protocols were formally verified by AVISPA, a commonly used tool for security protocol assessments. The entities and message exchanges were described by the HLPSL (High Level Protocol Specification Language) language [81].

AVISPA has four protocol validation modes called “Back ends”, including On the Fly Model Checker (OFMC) and CL-AtSe (Constraint Logic Based Attack Searcher). The results of the verification of a protocol are "SAFE", if no problem has been detected, and "UNSAFE", if an attack has been successful.

4.5.3.1. Modeling of the Proposed Protocols PROT_1 and PROT_2 in HLPSL

The protocol must be modeled according to HLSPSL for its evaluation by AVISPA. Figures 23, 24 and 25 show some parts of the modeling of the proposed protocols.

Figure 23 displays the modeling of EV behavior in HLPSL code. The following parts must be considered for the modeling of any entity in HLPLS: statement of the agents, communication channels, functions to be used, declaration of variables calculated or received by other entities, and constants known by the entity.

After the establishment of the entity's information, the operations and exchanges of messages between entities through states are described. Authentication variables and variables considered confidential are defined at the end of each state.

Proposed Protocol PROT_1	Proposed Protocol PROT_2
role role_EV(EV:agent,FS:agent,RSU:agent,PAD:agent, H1:function,H2:function,H3:function,H4:function, H5:function,CK:function,Kfsev:symmetric_key, Krsuev:symmetric_key,SND,RCV:channel(dy)) played_by EV def= local State:nat,T5:text,Sigfs:text,T6:text,Vfifs:text,Vfiev:text, C:text,PID:text,S:text,T7:text,Tao:text,T8:text,Y:text, PID2:text,T10:text,HMAC:function,Sigrsu:text,T11:text, Alf1:text,M:function,Alf2:text,P:text,Req:text,T12:text, T13:text,Is:text,Psi:text init State := 0 transition 1. State=0 /\ RCV(start) =|> State':=1 /\ T5':=new() /\ P':=new() /\ Vfiev':=new() /\ secret(Vfiev',sec_5,{}) /\ SND(M(Vfiev'.P').T5'.H1(M(Vfiev'.P').T5')) State=1 /\ RCV(M(Vfifs'.P).T6'.CK(M(Vfifs'.M(Vfiev.P))).Sigfs') =|> State':=2 /\ secret(Vfiev',sec_5,{}) /\ secret(Vfifs',sec_6,{}) /\ T7':=new() /\ C':=new() /\ S':=new() /\ PID':=new() /\ SND({PID'.S'.C'.T7'}_Kfsev) 4. State=2 /\ RCV({Alf1'.Tao'.PID2'.T8'}_Kfsev) =|> State':=3 /\ secret(Alf1',sec_1,{}) /\ T10':=new() /\ Y':=new() /\ SND(PID2'.Y'.T10'.HMAC(PID2'.Y'.T10'.Alf1')) 7. State=3 /\ RCV(M(H3(Alf2').P).T11'.CK(M(Alf1.M(Alf2'.P))).Sigrsu') =|> State':=4 /\ witness(EV,RSU,auth_10,Sigrsu') /\ secret(Alf2',sec_2,{}) /\ secret(Alf1',sec_1,{}) /\ T12':=new() /\ Req':=new() /\ SND({Req'.T12'}_Krsuev) 9. State=4 /\ RCV({Psi'.Is'.T13'}_Krsuev) =|> State':=5 /\ secret(Is',sec_4,{}) /\ secret(Psi',sec_3,{}) /\ SND(H5(Psi'.Is')) end role	role role_EV(EV:agent,FS:agent,RSU:agent,PAD:agent,H1:has h_func,Kfsev:symmetric_key,Krsuev:symmetric_key,Beta:t ext,SND,RCV:channel(dy)) played_by EV def= local State:nat,T1:text,Eta:text,T2:text,VSifs:text,VSiev:te xt,O:text,Teta:text,T:text,T3:text,Tao:text,T4:text,T6:text,T 7:text,Delta1:text,Cheby:hash_func,Delta2:text,HMAC:has h_func,Lambda:text,Psi:text init State := 0 transition 1. State=0 /\ RCV(start) =|> State':=1 /\ T1':=new() /\ VSiev':=new() /\ SND(Cheby(VSiev'.Beta).T1'.H1(Cheby(VSiev'.Beta).T1')) 2. State=1 /\ RCV(Cheby(VSifs'.Beta).T2'.H1(Cheby(VSifs'.Cheby(VSie v.Beta))).Eta') =|> State':=2 /\ T3':=new() /\ O':=new() /\ T':=new() /\ Teta':=new() /\ SND({Teta'.T'.O'.T3'}_Kfsev) 4. State=2 /\ RCV({Delta1'.Delta2'.Tao'.T4'}_Kfsev) =|> State':=3 /\ witness(EV,RSU,auth_8,Delta2') /\ secret(Delta2',sec_2,{}) /\ secret(Delta1',sec_1,{}) /\ T6':=new() /\ SND(H1(Delta1').T6'.HMAC(H1(Delta1').T6'.Delta2')) 7. State=3 /\ RCV(Cheby(H1(Delta2').Beta).T7'.H1(Cheby(H1(Delta1'). Cheby(H1(Delta2').Beta))).{Psi'.Lambda'}_Krsuev.HMAC(Cheby(H1(Delta2').Beta).T7'.H1(Cheby(H1(Delta1').Cheby (H1(Delta2').Beta))).{Psi'.Lambda'}_Krsuev)) =|> State':=4 /\ secret(Lambda',sec_4,{}) /\ secret(Psi',sec_3,{}) /\ SND(H1(Psi'.Lambda')) end role

FIGURE 23. ROLE OF EV IN HLPSL

Figure 24 shows the protocols execution environment, session establishment and elements that can be acquired by the attacker, and the definition in HLPSL language code.

Proposed Protocol PROT_1	Proposed Protocol PROT_2
role session1(Krsuev:symmetric_key,HMAC:function,KGfsrsu:symmetric_key,CK:function,Kfsev:symmetric_key,EV:agent,FS:agent,RSU:agent,PAD:agent,H1:function,H2:function,H3:function,H4:function,H5:function,KGrsupad:symmetric_key) def= local SND4,RCV4,SND3,RCV3,SND2,RCV2,SND1,RCV1:channel(dy) composition role_PAD(EV,FS,RSU,PAD,H1,H2,H3,H4,H5,KGrsupad,SND4,RCV4) /\ role_RSU(EV,FS,RSU,PAD,H1,H2,H3,H4,KGfsrsu,HMAC,Krsuev,KGrsupad,SND3,RCV3) /\ role_FS(EV,FS,RSU,PAD,H1,H2,CK,Kfsev,KGfsrsu,HMAC,SND2,RCV2) /\ role_EV(EV,FS,RSU,PAD,H1,H2,H3,H4,H5,CK,Kfsev,Krsuev,SND1,RCV1) end role	role session1(Krsuev:symmetric_key,HMAC:hash_func,KGfsrsu:symmetric_key,Kfsev:symmetric_key,EV:agent,FS:agent,RSU:agent,PAD:agent,H1:hash_func,KGrsupad:symmetric_key,Beta:text) def= local SND4,RCV4,SND3,RCV3,SND2,RCV2,SND1,RCV1:channel(dy) composition role_PAD(EV,FS,RSU,PAD,H1,KGrsupad,Beta,SND4,RCV4) /\ role_RSU(EV,FS,RSU,PAD,H1,KGfsrsu,HMAC,Krsuev,KGrsupad,Beta,SND3,RCV3) /\ role_FS(EV,FS,RSU,PAD,H1,Kfsev,KGfsrsu,HMAC,Beta,SND2,RCV2) /\ role_EV(EV,FS,RSU,PAD,H1,Kfsev,Krsuev,Beta,SND1,RCV1) end role

Figure 24. HLPSL codification of the role session

Finally, Figure 25 displays the security objectives of the proposed protocols, which depend on the variables defined as secret and authentication values defined in the roles of the entities.

Proposed Protocol PROT_1	Proposed Protocol PROT_2
goal secrecy_of sec_1 secrecy_of sec_2 secrecy_of sec_3 secrecy_of sec_4 secrecy_of sec_5 secrecy_of sec_6 authentication_on auth_7 authentication_on auth_8 authentication_on auth_9 authentication_on auth_10 authentication_on auth_11 end goal	goal secrecy_of sec_1 secrecy_of sec_2 authentication_on auth_3 authentication_on auth_4 authentication_on auth_5 authentication_on auth_6 authentication_on auth_7 end goal

FIGURE 25. SECURITY OBJECTIVES AND RELATED SECRETS OF THE PROTOCOLS PROT_1 AND PROT_2 IN HLPSL

The security objectives of the Proposed Protocol PROT_1 are:

- secrecy_of sec_1: keep secret α_1;
- secrecy_of sec_2: keep secret α_2
- secrecy_of sec_3: keep secret ψ
- secrecy_of sec_4: keep secret υ
- secrecy_of sec_5: keep secret ϕ_{EV}
- secrecy_of sec_6: keep secret ϕ_{fs}
- authentication_on auth_7: EV authenticates FS on σ_{fs};
- authentication_on auth_8: FS authenticates EV on PID_1;
- authentication_on auth_9: RSU authenticates EV on α_1;
- authentication_on auth_10: EV authenticates RSU on $H(\alpha_2)$;
- authentication_on auth_11: Pad authenticates EV on υ;

On the other hand, the security objectives of the Proposed Protocol PROT_2 are:

- secrecy_of sec_1: keep secret ψ
- secrecy_of sec_2: keep secret λ
- authentication_on auth_3: EV authenticates FS on θ;
- authentication_on auth_4: FS authenticates EV on ψ;
- authentication_on auth_5: RSU authenticates EV on σ_{rsu_1};
- authentication_on auth_6: EV authenticates RSU on δ_2;
- authentication_on auth_7: Pad authenticates EV on v;

4.5.3.2. Security Check Results

The safety of the proposed protocols was confirmed by a simulation in AVISPA using CL-AtSe and OFMC back ends. (see Fig. 26).

Backend	CL-AtSe	OFMC
Proposed Protocol PROT_1	SUMMARY SAFE DETAILS BOUNDED_NUMBER_OF_SESSIONS TYPED_MODEL PROTOCOL /home/span/span/testsuite/results/hlpslGenFile.if GOAL As Specified BACKEND CL-AtSe STATISTICS Analysed : 27 states Reachable : 8 states Translation: 0.44 seconds Computation: 0.00 seconds	% OFMC % Version of 2006/02/13 SUMMARY SAFE DETAILS BOUNDED_NUMBER_OF_SESSIONS PROTOCOL /home/span/span/testsuite/results/hlpslGenFile.if GOAL as_specified BACKEND OFMC COMMENTS STATISTICS parseTime: 0.00s searchTime: 0.24s visitedNodes: 11 nodes depth: 6 plies
Proposed Protocol PROT_2	Subject: SUMMARY SAFE DETAILS BOUNDED_NUMBER_OF_SESSIONS TYPED_MODEL PROTOCOL /home/span/span/testsuite/results/chaos.if GOAL As Specified BACKEND CL-AtSe STATISTICS Analysed : 10 states Reachable : 7 states Translation: 0.20 seconds Computation: 0.00 seconds	% OFMC % Version of 2006/02/13 SUMMARY SAFE DETAILS BOUNDED_NUMBER_OF_SESSIONS PROTOCOL /home/span/span/testsuite/results/chaos.if GOAL as_specified BACKEND OFMC COMMENTS STATISTICS parseTime: 0.00s searchTime: 0.15s visitedNodes: 11 nodes depth: 6 plies

FIGURE 26. SECURITY SIMULATION RESULTS FOR CL-ATSE AND OFMC BACKENDS

4.6. Summary

This chapter addressed the design and operation of the two proposed protocols in a centralized model. The first protocol (PROT_1) was created from a cryptographic scheme based on bilinear pairing, elliptical curves, and hash chains and showed lower computational costs and a more complete security analysis compared to other schemes. The design of the second (PROT_2) was based on chaotic cryptography for authentication and access control in a CWD-WPT charging system that uses fog servers to optimize system latency times and user travel times by recharging the battery with magnetic induction while the vehicle is in motion. The Protocol uses new cryptographic primitives based on chaotic maps (e.g., digital signature, blind signature, and key agreement), which have low computational costs compared to cryptographic primitives based on bilinear pairing. It also employs other cryptographic primitives such as HMACs and hash chains. Compared to other schemes, it imposed lower computational costs and the security analysis was more complete in terms of security properties and protection against attacks.

5. PROTOCOL FOR DECENTRALIZED CWD-WPT CHARGING STATION

In this chapter, a decentralized network model is considered, since the system is managed locally by several Charging Control Centers close to the highway where the pads are installed. A protocol (PROT_3) was designed for this model considering the adversary model described in Section 4.2.

5.1. Decentralized CWD-WPT Charging Station

A decentralized CWD-WPT System supported by a private blockchain system and Redundant Byzantine Fault Tolerance (RBFT) as a consensus method was considered. The network model is composed of the following elements, as shown in figure 27:

- Trusted Authority (TA): installed on the cloud, it registers and generates system and user keys and creates the genesis block of the blockchain system;
- Charging Control Center (CCC) : a station that controls the station's charging pads;
- RSUs (road side units): units deployed at the margins of the road and implemented by access points;
- Fog Server (FS): installed near the RSUs and the CCC, it creates and checks the blockchain blocks in the system;
- A group of "ψ" charging pads {PD_1, PD_2, …, $PD\psi$} installed on the floor of the charging station, they induce an electric charge in the moving EVs served by the CWD-WPT system; and
- Electrical vehicles (EVs).

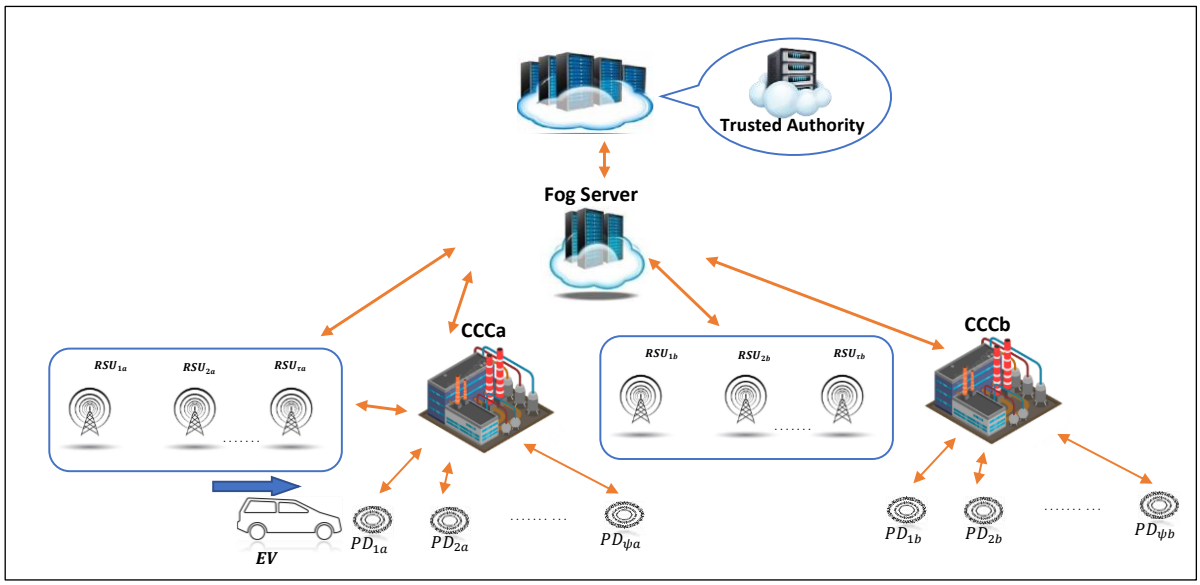


FIGURE 27. NETWORK MODEL OF A DECENTRALIZED CWD-WPT CHARGING STATION

The blockchain system is supported by TA, where the genesis block is generated and the FSs receive messages and create the blocks that contain the transactions. Towards a performance comparison among other protocols and the proposed one, according to [60] and [73], a charging station can be 4.2 km long and is managed by 1 CCC, 7 RSUs positioned 600 meters apart from each other, and each CCC manages 5250 pads separated by 40 centimeters. Table 15 shows the characteristics of the charging station considered for this architecture.

Table 15. Characteristics of the charging station´s decentralized architecture

Entities	CCC	FS	RSUs (τ)	Pads (ψ)
Number of entities for Charging Stations	1	1	7	5250
Separation Between Entities of the Same Type	N/A	N/A	600 m	40 cm

5.2. PROT_3 - Chaotic Map- and blockchain-based authentication protocol for CWD-WPT charging system

Initially, the system must choose the functions and generate the keys; then, the FS, CCC, RSUs, pads, and EVs entities are registered by the system. During this process, the identification keys and other variables to be further used are assigned and the blockchain system whose operations are supported by FS are established. Users can buy tickets offline during EV registration to use the CWD-WPT charging station.

The group of entities that supported the operation of the CWD-WPT load service is then generated. Next, an EV that intends to use the CWD-WPT charging station can communicate with the nearest RSU to start the authentication and recharge process.

The proposed protocol PROT_3 has the following 5 phases (fig.28):

1: System Initialization;
2: Registration of entities (FS, RSUs, pads and EVs)
3: Registration of EVs and Ticket Purchasing;
4: EV Authentication;
5: Charging Request.

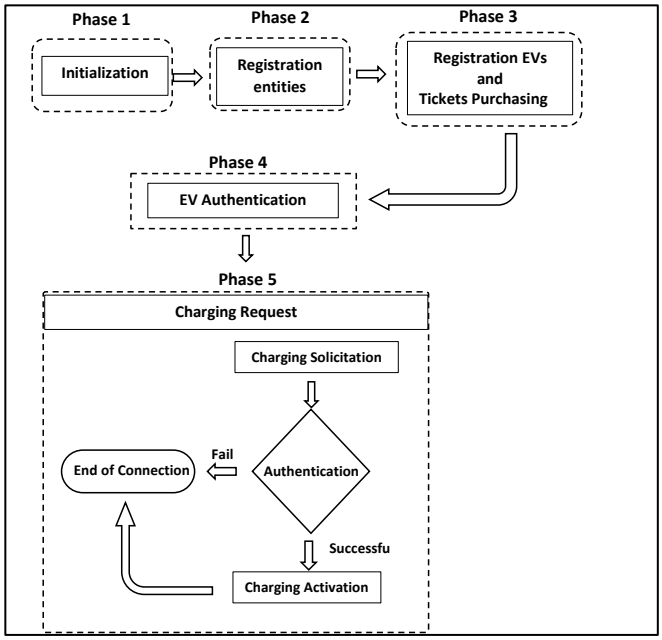


FIGURE 28. PHASES OF THE PROPOSED PROTOCOL PROT_3

In the proposed protocol, the CCC is considered safe, FS, RSUs and pads are considered safe but curious, and EVs are considered unsafe. On the other hand, communications that support the functioning of the system, but are not directly part of the authentication or access control processes, which are the focus of this thesis, are therefore assumed to be secure, that is, communications are secure between:

- the FS and the RSUs, on all phases,
- RSUs and pads, on all phases,
- the EVs and the CCC, on the registration and purchase of tickets phases.

Communications are insecure between:

- the EVs and the FS in the EV authentication and charging request phases,
- the EVs and the RSU in the EV authentication and charging request phases,
- the EVs and pads in the EV authentication and charging request phases.

The phases of the PROT_3 protocol are described in what follows.

1st phase: **System Initialization**

Let $\boldsymbol{p}$ be a **large prime number** and $\boldsymbol{n}$ a factor of $p-1$ and the product of two random prime numbers $\bar{p}$ and $\bar{q}$ ie $n = \bar{p}\bar{q}$. Let β be an element of an finite group of order module $\boldsymbol{n}$ and a generator element of the multiplicative group of set $\boldsymbol{G}$ [48].

The system chooses a random number $\boldsymbol{e} \in Z_p^*$ such that $gcd\,(e,n) = 1$, and a number $\boldsymbol{d}$ such that $e.d = 1\,(mod\ \varphi(n))$, where $\varphi(n) = (\bar{p}-1)(\bar{q}-1)$. The hash function of the system is defined as $H: \{0,1\}^* \rightarrow \mathbb{Z}_p^*$.

The Trusted Authority (TA) then chooses a master private key $s_{ta}, \in Z_q^*$ and calculates its global public key $Y_{pub} = T_{s_{ta}}(\beta) mod(p)$, where $\boldsymbol{T}$ is the Chebyshev polynomial map of degree β defined as the following recurrent function[45]:

$$T_{s_{ta}}(\beta) = \cos\left(s_{ta} * arcos(\beta)\right) \begin{cases} T_0(\beta) = 1\, mod(p),\ for\, s_{ta} = 0; \\ T_1(\beta) = \beta\, mod(p),\ for\, s_{ta} = 1; \\ T_{s_{ta}}(\beta) = (2\rho\, T_{s_{ta}-1}(\beta) - T_{s_{ta}-2}(\beta))\, mod(p),\ for\, s_{ta} > 1 \end{cases} \quad (91)$$

Finally, the TA publishes the genesis block with parameters $Tx\{(H, p, \beta, Y_{pub}, T, E/D\ protocols), H, p, \beta, Y_{pub}, T, E/D\ protocols\}$ for blockchain, where E/D are the encryption and decryption protocols, and master key $\boldsymbol{s_{ta}}$ are kept secret.

2nd phase: **Registration entities and group creation**

Registration entities:

The following process is conducted for the registration of FS/CCC/RSU/Pads:

The entity sends its ID_i entity to the TA through a secure channel

$$m_1 = \{ID_i\}$$

The TA receives the message and generates a random number ϕ, private key $x_i = h(\phi, ID_i)$, and public key $y_i = T_{x_i.s}(\beta) = T_{x_i}(Y_{pub})$. After calculating the keys, it sends the device a message m_2 containing the calculated keys, a group identifier, and the device's L position in the group list.

$$m_2 = \{x, y, ID_{Gi}\}$$

The device generates a random number a_i, calculates $A_i = T_{a_i}(\beta)$, and sends a message (m_3) to the TA.

$$m_3 = \{A_i, y_i\}$$

The TA then sends a message $m_4 = Tx(h(A_i, y_i), A_i, y_i)$ to the Blockchain. Figure 29 shows a diagram of messages exchanged in the registration phase

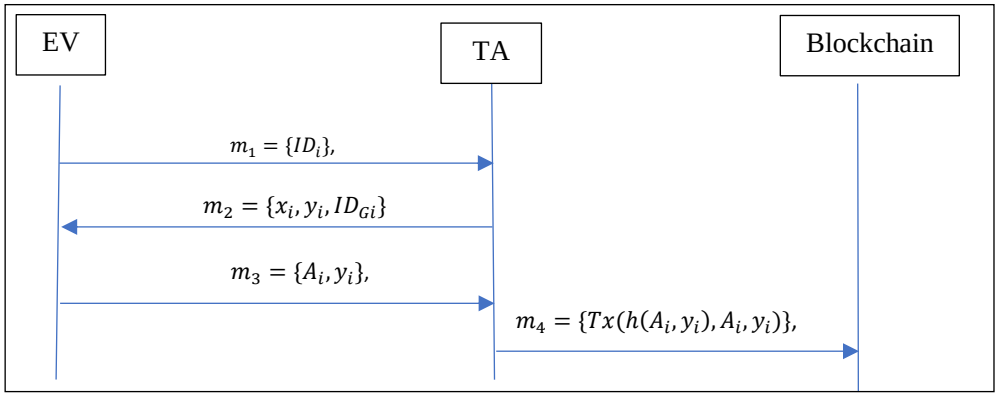


FIGURE 29. REGISTRATION ENTITIES (PROPOSED PROTOCOL PROT_3)

Group creation:

a) Each member of group G_i generates a random number m_i, takes an A_{i+1} from the blockchain, and calculates the following values: $M_i = T_{m_i}(\beta)$; $k_{B_{i+1}} = T_{a_i}(A_{i+1})$; $SE_{i+1} = \{M_i\}_{E_{k_{B_i+1}}}$; $C_i = T_{h\left(SE_{i+1}, k_{B_{i+1}}, ts_{1_i}\right).x_i}(\beta)$, where ts_{1_i} is the timestamp.

b) Device Dev_i sends a message m_{1_i} to device Dev_{i+1}

$$m_{1_i} = \{SE_{i+1}, C_i, ts_{1_i}\}$$

c) When Dev_i receives message $m_{1_{i-1}} = \{SE_i, C_{i-1}, ts_{1_{i-1}}\}$ from Dev_{i-1}, it checks timestamp $ts_{1_{i-1}} < \Delta t$. If the check succeeds, it calculates $k_{B_i} = T_{a_i}\left(T_{a_{i-1}}(\beta)\right) = T_{a_i.a_{i-1}}(\beta)$ and then decrypts message SE_{i-1} sent by Dev_{i-1}.

$$M_{i-1} = \{SE_{i-1}\}_{D_{k_{B_i}}}$$

d) Dev_i performs the following verification to authenticate Dev_{i-1}

$$\alpha = h\left(SE_i, k_{B_{i-1}}, ts_{1_{i-1}}\right), \quad (92)$$

$$C_i(Y) =? T_\alpha(y_{i-1}). \quad (93)$$

Mathematical validation:

$$C_i(Y) = T_{h\left(SE_i, k_{B_{i-1}}, ts_{1_{i-1}}\right).x_{i-1}}(Y), \tag{94}$$

$$C_i(Y) = T_{h\left(SE_i, k_{B_{i-1}}, ts_{1_{i-1}}\right).x_{i-1}}(T_s(\beta)), \tag{95}$$

$$C_i(Y) = T_{h\left(SE_i, k_{B_{i-1}}, ts_{1_{i-1}}\right)}\left(T_{s.x_{i-1}}(\beta)\right), \tag{96}$$

$$C_i(Y) = T_{h\left(SE_i, k_{B_{i-1}}, ts_{1_{i-1}}\right)}(y_{i-1}), \tag{97}$$

$$C_i(Y) = T_\alpha(y_{i-1}). \tag{98}$$

If the validation is successful, Dev_i recognizes device Dev_{i-1} as a valid member of the group. The check is similar to the one performed by Dev_{i+1} with Dev_i

e) Dev_{i+1} calculates key $k_{B_{i+1}}$ and decrypts the message sent by Dev_i

$$M_i = \{SE_{i+1}\}_{D_{k_{B_{i+1}}}}. \tag{99}$$

f) Then Dev_{i+1} returns a message $m_{2_i} = \left\{M_i, ID_{G_i}\right\}_{D_{k_{B_{i+1}}}}$ to Dev_i

g) Dev_i validates m_{2_i} content $M_i =? M_i'$ and Dev_{i+1} recognizes device Dev_i as a valid member of the group.

h) The devices then calculate a check value $R_i = T_{h\left(ID_{G_i}\right)}(y_i)$. Finally, a broadcast message is sent to the other group members. $\{R_i, y_i\}$. Figure 30 shows the message exchange for the creation of the device group.

i) After receiving messages from the other group members, each device group verifies their authenticity.

$$\prod_1^n R_i =? T_{h\left(ID_{G_i}\right)}(y_1).T_{h\left(ID_{G_i}\right)}(y_2).\ldots\ldots.T_{h\left(ID_{G_i}\right)}(y_n). \tag{100}$$

If the verification is successful, the device calculates temporary group key k_t

$$k_t = T_{h(\prod_1^n R_i)}(\beta). \tag{101}$$

j) With this temporary key, each group member sends the previously calculated variable to the other group members $\{M_i\}_{k_t}$, which calculate the definitive group key:

$$\psi = h(M_1, M_2, \ldots., M_n) \tag{102}$$

$$k_{G_i} = T_\psi(k_t) \tag{103}$$

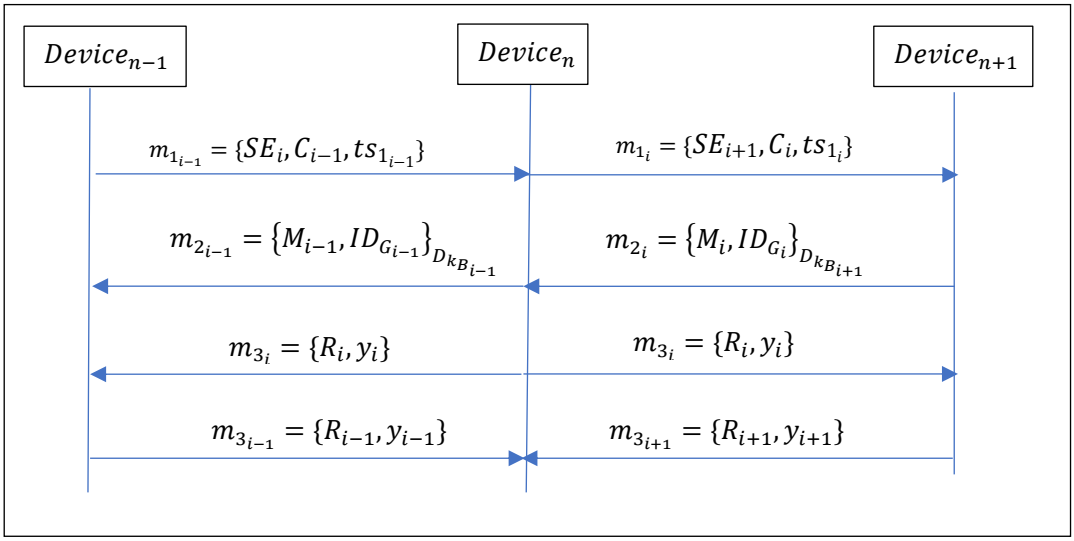


FIGURE 30. GROUP CREATION (PROPOSED PROTOCOL PROT_3)

Device joining the group:

The following sequence describes the way a new device (RSU, FS, Pad) can be added to the group:

a) The new $NDev$ device is registered in the TA and the identity of group ID_{G_i} where it will be added is chosen. The TA sends the Blockchain a message with the update of the list of devices that are part of the group. The Blockchain then provides $NDev$ with a list of devices that form the group to which it will be added.

b) $NDev$ is positioned as the last device in the list, sends a message $m_1 = \{SE_1, C_{n+1}, ts_{1_{n+1}}\}$ to the first device in the Dev_1 list, and receives a message $m_2 = \{SE_{n+1}, C_n, ts_{1_n}\}$ from the last device in the Dev_n list.

c) Dev_1 and Dev_n follow steps c), d), e), f), and g) of the group creation section. If the validation of the new group member by Dev_1 and Dev_n is successful, the new member $NDev_{n+1}$ calculates $R_{n+1} = T_{h(ID_{G_i})}(y_{n+1})$. Finally, a broadcast message message $m_3 = \{R_{n+1}, y_{n+1}\}$is sent to the other group members, which then check its authenticity.

$$\prod_{1}^{n+1} R_i = ? T_{h(ID_{G_i})}(y_1).T_{h(ID_{G_i})}(y_2).\ldots\ldots.T_{h(ID_{G_i})}(y_{n+1}). \quad (104)$$

If the verification is successful, the devices in the group calculate the new group key

$$k_{G_i} = T_{\psi}(R_i). \quad (105)$$

$$k_{G_i} = T_{R_i.\psi}(\beta). \quad (106)$$

d) Finally, Dev_1 and Dev_n send the new group key $\{k_{G_i}\}$ to Dev_{n+1} in an encrypted form.

Device leaving the group:

When one of the devices (FS/RSU/Pad) with identity Dev_j must leave the group, the group performs the following steps:

a) The TS sends a message to the Blockchain confirming the disconnection of Dev_j from the group with $ID_{G_i}, Tx(h(M_j, y_j, desable), M_j, y_j, desable)$.

b) The group members choose a new $M_i' = T_{m_i'}(\beta)$ and again perform the procedure described in "**Group creation**" items b), c), d), f), g), h), i), j), l), and m) to obtain a new group key $k_{G_i}' = T_{\psi'}(k_t')$ without the device disabled.

3rd phase: **Registration of EVs and purchase of tickets**

The process described below is followed for the registration of EVs:

The EV owner sends the TA vehicle identity ID_{ev} , personal information, and the bank account.

$$m_1 = \{ID_{ev}, Personal\ data, bank\ account\}$$

The TA generates a random number ϕ_{ev}, private key $x_{ev} = h(\phi_{ev}, ID_{ev})$, and public key $y_{ev} = T_{x_{ev}.s}(\beta) = T_{x_{ev}}(Y_{pub})$ and calculates check values $V_{1ev} = h(PID_{ev}, y_{ev})$ and $V_{2ev} = h(ID_{ev}, x_{ev})$. It then sends the EV a message m_2

$$m_2 = \{x_{ev}, y_{ev}, PID_{ev}, V_{2ev}\}$$

and a message $m_3 = Tx(h(y_{ev}, PID_{ev}),\ y_{ev}, PID_{ev})$ to the Blockchain.

The EV stores the values to be used for its authentication to the network. Figure 31 shows the EV registration process

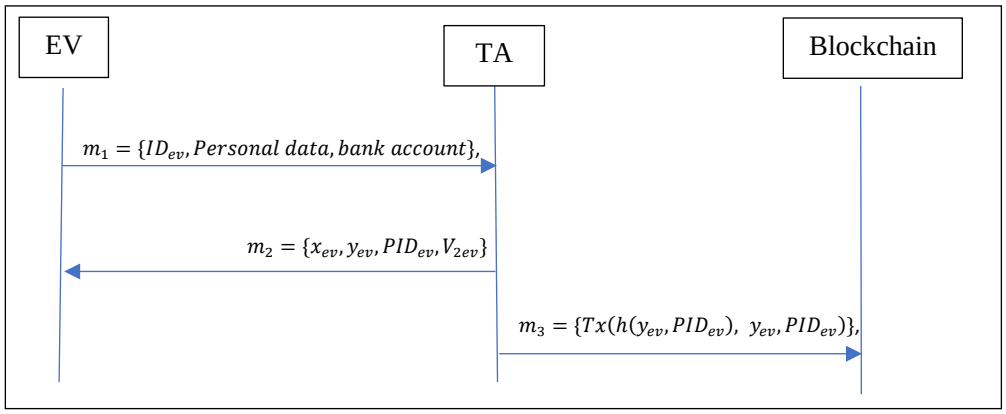


FIGURE 31. REGISTRATION OF EVS (PROPOSED PROTOCOL PROT_3)

Purchase of tickets:

The ticket purchase process is conducted by the EV owner offline. When the EV node wants to buy tickets to access the CWD-WPT charging service, it communicates with the TA through a secure channel and sends it a message m_1

$$m_1 = \{h(PID_{ev}, y_{ev}, z), y_{ev}, z\},$$

The TA checks the integrity and identity of the EV. If the verification is correct, the TA debits the value of the number of tickets from the associated bank account and generates z random numbers η_w where z corresponds to the number of tickets requested and w is a number between $1 \leq w \leq z$.

It then calculates tuple

$$k_w = T_{\eta_w}(\beta) \quad (107)$$
$$TK_w = T_{k_w}.s(\beta) = T_{k_w}(Y) \quad (108)$$

sends a message m_2 with the set of purchased tickets to the EV

$$m_2 = \{k_w, TK_w\}$$

and a message m_3 to the Blockchain with one of the tuple values of the generated tickets $Tx(h(TK_w, Y), TK_w, Y)$

When the EV receives m_2, it stores the tickets securely for a later use . Figure 32 shows the ticket purchase process

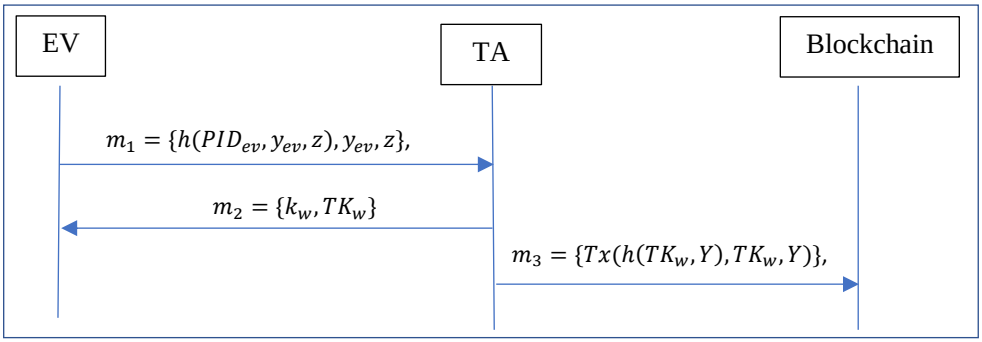


FIGURE 32. PURCHASE OF TICKETS (PROPOSED PROTOCOL PROT_3)

4th phase: **EV Authentication**

When the EV wants to use a charging station, it first authenticates itself with the closest RSU that is part of the group in which the CWD-WPT charging station is inserted.

The EV starts the authentication process against the RSU by sending it message m_1

$$m_1 = \{h(v_{1_{ev}}, y_{ev}, ts_1), v_{1_{ev}}, ts_1\}$$

The RSU checks timestamp $ts_1 \leq \Delta t$ and, through the blockchain, it checks if it is a registered user $v_{1_{ev}} =? v_{1_{ev}}'$. If no registration is found, the RSU closes the connection; otherwise, i.e., if the verification is successful, it verifies the integrity of the message by comparing the hash that arrived in it with the hash calculated with values published in the Blockchain.

$$h(v_{1_{ev}}, y_{ev}, ts_1) =? h(v_{1_{ev}}', y_{ev}', ts_1') \quad (109)$$

If the verification is successful, the RSU selects two random values τ and ε and calculates the session key with τ .

$$k_s = T_{\tau.x_{rsu}}(y_{ev}) = T_{\tau.\ x_{rsu}.x_{ev}}(Y) \quad (110)$$

RSU sends message m_2 containing value ε encrypted with key k_s, a key verification value, and other values to the EV

$$m_2 = \{\{\varepsilon\}_{k_s}, h(k_s), \tau, ts_2, h(\{\varepsilon\}_{k_s}, h(k_s), \tau, ts_2)\}$$

The EV checks timestamp $ts_2 \leq \Delta t$ and the integrity of the message by checking the hash. If the checks are successful, it then calculates the session key using its private key, the $\boldsymbol{\tau}$ that arrived in the message, and the public key from the RSU:

$$k'_s = T_{\tau.x_{ev}}(y_{rsu}) = T_{\tau.x_{ev}.x_{rsu}}(Y) \quad (111)$$

It checks if the key is correct by comparing it with the verification value that arrived in m_2, i.e., if $h(k'_s) =? h(k_s)$. If the comparison is successful, it authenticates the RSU as valid , the EV decrypts message $\{\varepsilon\}_{k_s}$ to obtain ε, and sends m_3 to the RSU

$$m_3 = \{h(\varepsilon, \tau, k_s, ts_3), ts_3\}$$

The RSU checks timestamp $ts_3 \leq \Delta t$ and then compares the hash that arrived in the message with the hash it calculated.

$$h(\varepsilon, \tau, k_s, ts_3) =? h'(\varepsilon, \tau, k_s, ts'_3) \quad (112)$$

If the comparison is successful, the RSU authenticates the EV and then sends a message encrypted with group key $m_4 = \{PID, \varepsilon\}_{k_{G_i}}$ to the group members and a message m_5 *to* the Blockchain.

$$m_5 = Tx(h(v_1, h(\varepsilon), y_{ev}), v_1, h(\varepsilon), y_{ev})$$

The messages aim at speeding up the EV authentication in the next RSUs. Figure 33 shows the message exchange performed by the protocol in the EV authentication process.

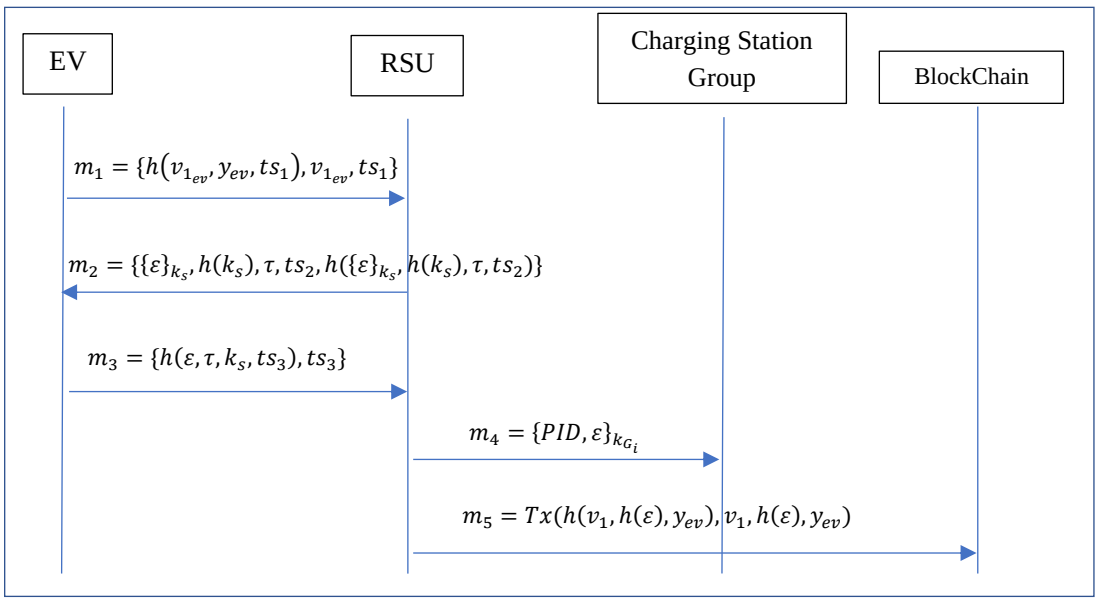


FIGURE 33. EV AUTHENTICATION (PROPOSED PROTOCOL PROT_3)

5th phase: **Charging Request**

When the EV approaches the charging station, it sends by CCC an m_1 message to authenticate itself.

$$m_1 = \{h(\varepsilon, v_{1_{ev}}, y_{ev}, ts_1), v_{1_{ev}}, y_{ev}, ts_1\}$$

The CCC validates the information in m_1 with the values sent by the group's RSU and compares the message hash (h) with the hash calculated (h') .

$$h(\varepsilon, v_{1_{ev}}, y_{ev}, ts_1) = h'(\varepsilon, h(PID_{ev}, y_{ev}), y_{ev}', ts_1') \quad (113)$$

If the validation is successful, the CCC selects a random number $\boldsymbol{u}$, calculates a session key $k_{st} = T_{u.\varepsilon}(\beta)$, and sends the hash of this key to the EV along with the random value u in message m_2

$$m_2 = \{h(T_{u.\varepsilon}(\beta), u, ts_2), u, ts_2\}$$

The EV checks timestamp $ts_2 \leq \Delta t$, calculates session key $k_{st} = T_{\varepsilon.u}(\beta)$ with value u, and compares the message hash (h) and the hash calculated by (h').

$$h(T_{u.\varepsilon}(\beta), u, ts_2) =? h(T_{u.\varepsilon}(\beta)', u, ts_2') \quad (114)$$

If the comparison is successful, the EV authenticates the CCC and verifies the message integrity; otherwise, it closes communication. Once a session key has been established, the EV sends the ticket to use the charging station to the CCC in message m_3.

$$m_3 = \{\{k_w, TK_w\}_{k_{st}}, ts_3\}$$

The CCC checks timestamp $ts_3 \leq \Delta t$, decrypts the message, and get the ticket (k_w, TK_w). It checks the authenticity of the ticket calculating

$$T_{k_w}(Y) = TK_w \quad (115)$$

If the ticket is authentic, the CCC checks the blockchain on the use of the ticket. If it is a used one, the CCC alerts the EV the ticket is not valid; otherwise, i.e., if the ticket is valid, the CCC authorizes the use of the CWD-WPT charging station and sends a message m_4 for the EV with a seed λ for the calculation of the authentication keys of the pads and the number of pads $\boldsymbol{\psi}$ that controls the charging station. Simultaneously, it sends a message m_5 encrypted with the group key and seed λ to the pads.

$$m_4 = \{\lambda, \boldsymbol{\psi}\}_{k_{st}}$$
$$m_5 = \{\lambda, \boldsymbol{\psi}\}_{k_{G_i}}$$

After receiving and decrypting the m_4 and m_5, the messages by the EV and the pads respectively, they calculate the authentication keys using a hash string.

$$k_{p_{\psi+1}} = h(\lambda);\ k_{p_\psi} = h\left(k_{p_{\psi+1}}\right);\ k_{p_{\psi-1}} = h\left(k_{p_\psi}\right) \ldots\ldots k_{p_2} = h(k_{p_3});\ k_{p_1} = h(k_{p_2}) \quad (116)$$

k_{p_2} is the first key to be used and $k_{p_{\psi+1}}$ is the last. k_{p_1} is a check value. Each pad selects a switch according to its position in the charging station.

When the EV sends the authentication key corresponding to each pad, the pad verifies the authenticity against the key it has assigned. The pad will activate if the key is authentic. When the key has been successfully used, the pad sends a broadcast message with such a key to be added to a revocation list. If

the pad finds the key sent by the EV does not match the assigned one, the EV is treated as valid and the pad does not induce energy. If it is valid, but the pad verifies it is on the revocation list, it does not induce power to the EV. Figure 34 shows the proposed charging request process described above.

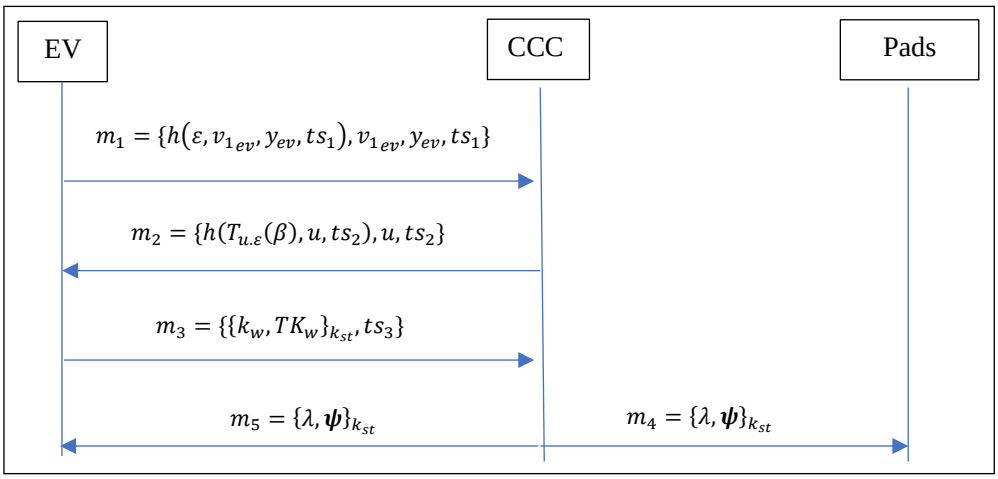


FIGURE 34. CHARGING REQUEST (PROPOSED PROTOCOL PROT_3)

5.2.1.Performance Analysis

Below is an analysis and a performance comparison of the communication, computational, and energy costs of the proposed protocol with those of other schemes.

5.2.1.1. Communication Costs

The total number of bytes transmitted by a network during the execution of the protocol is considered for the calculation of the communication cost, since it involves the number of data transmitted directly in the bandwidth for the operation of the protocol. However, the headers or control bits inherent to the communication protocol used are not considered. The quantity of bytes of each parameter adopted in each message, size of each message with its parameters, and number of messages are considered.

Table 16 shows the values in bytes of each variable (values taken from Rabieh and Wei [25]) .

TABLE 16. SYMBOLS AND COSTS IN BYTES (PROPOSED PROTOCOL PROT_3)

Symbol	Description	Length (Bytes)
ID	Identification	128
PID	Pseudo Identity	32
$H()$	Hash function	32
Tx	blockchain transaction	108
x, S	Private key	32
Y,y	Public key	32
k	Session key	32
(k_w, TK_w)	Ticket	64
ψ	Number of pads for RSU	8
λ	Seed	20
ts	Timestamp	8
$p, n, u, \tau, \varepsilon,$	Prime numbers	32
HMAC	Hash-based message authentication code	32

$\boldsymbol{n}$ EVs, $\boldsymbol{\tau}$ RSUs and $\boldsymbol{\psi}$ pads (for each RSU) were considered for the calculation of the communications cost of the protocol. Table 17 shows a comparison of the costs among our protocol and those of Pazos-Revilla et al.[58] and L. Roman and P. Gondim [82].

TABLE 17. COMMUNICATION COSTS IN BYTES (PROPOSED PROTOCOL PROT_3)

Message	Pazos-Revilla et al. [58]	L. Roman and P. Gondim [82]	Proposed Protocol
M1	$32n$	$74n$	$74n$
M2	$128n$	$74n$	$116n$
M3	$128n$	$104n$	$40n$
M4	$96n$	$96n$	$64n$
M5	$32n$	$64n$	$108n$
M6	$40n$	$60n$	$104n$
M7	40τ	$40n$	$74n$
M8	$32(n*\tau)$	$32(n*\psi)$	$74n$
M9	$32(n*\tau)$	$74(n*\tau-1)$	$64(n+1)$
M10	$32(n*\tau*\psi)$	$68(n*\tau-1)$	$32(n*\psi)$
M11	$--$	$32(n*\tau-1)$	$--$
M12	$--$	$32(n*\tau-1*\psi)$	$--$
Total	$\boldsymbol{n(556+\tau 64+32\tau\psi)+40\tau}$	$\boldsymbol{n(552+206\tau+32\tau\psi)}$	$\boldsymbol{n(750+(32*1500))+64}$

According to Table 15, was considered for the comparison of our protocol with those of Pazos-Revilla et al.[58] and L. Roman and P. Gondim [82], respectively. Figure 35 shows the communication cost is similar to that of Pazos-Revilla et al.[58] and L. Roman and P. Gondim [82], since the highest communications cost for all protocols is incurred in the upload process, when the EV sends the authentication messages to the pads.

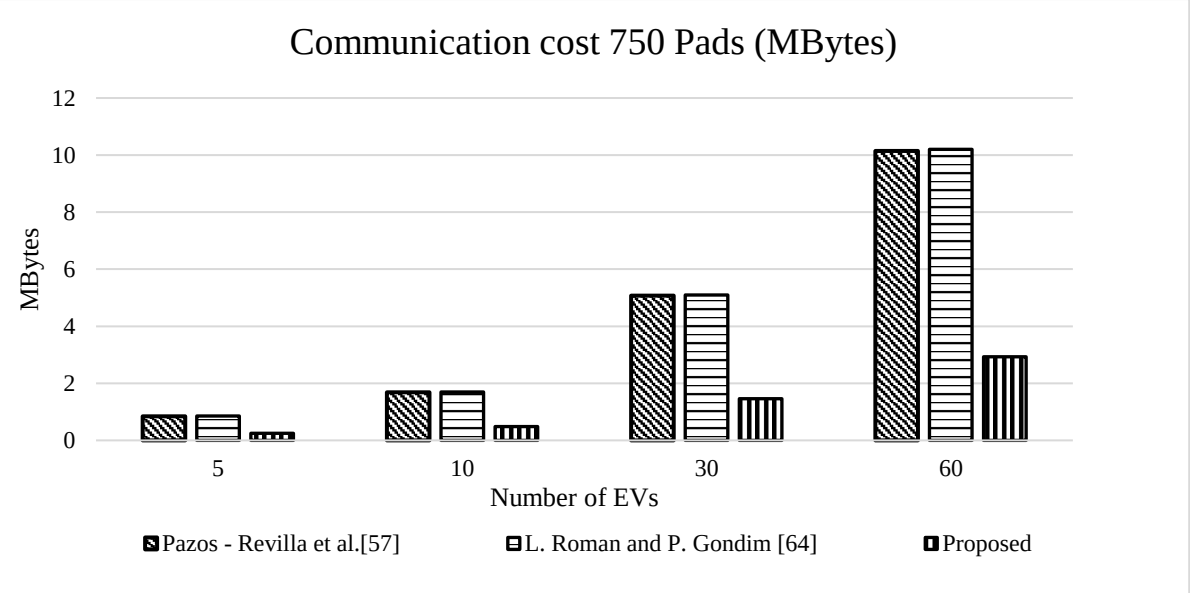


FIGURE 35. COMPARISON OF COMMUNICATION COSTS (PROPOSED PROTOCOL PROT_3) .

5.2.1.2. Computational Costs

An estimate of the time necessary for the execution of unitary operations that are part of the messages previously described in the phases of the protocol, as well as the differences among entities regarding the respective processing power are considered for the evaluation of computational costs. The cost values are based on common and realistic values obtained by experimentation and adopted for performance comparisons of authentication protocols.

Towards defining a reference architecture for the evaluation of authentication protocols, Tao et al. [52] obtained the computational cost of each unitary operation taking into account 3 hardware types:

- mobile equipment (processor Qualcomm (R) Octa core 1.5 GHz, 2G RAM).
- a Desktop (Intel (R) Dual core processor 3.1 GHz, 4GB RAM), and
- a Server (Intel (R) Hexa core processor 1.6 GHz and 16G RAM).

For our study, such hardware types correspond to EV/pad, RSUs, and FS/CSP, respectively.

TABLE 18. COSTS IN $\boldsymbol{ms}$ OF EACH OPERATION AND ENTITY (PROPOSED PROTOCOL PROT_3)

Entity	Costs (ms)				
	Tmp_{mul}	Tmp_{exp}	Tmp_{pair}	Tmp_{hash}	Tmp_{Chaos}
EV/Pad	0.29	0.5	0.75	$0.3x10^3$	0.95
RSUs	0.22	0.38	0.57	$0.2x10^3$	0.07
FS/ CSP	0.17	0.31	0.46	$0.1x10^3$	0.05

The methodology adopted for the performance evaluation considers the cost of each unitary operation multiplied by the number of times each operation is executed and the several messages that include one or more of such unitary operations, as required for the different authentication protocols. Table 18 shows the execution times of the cryptographic unitary operations used by the different protocols, according to the values provided in [24][83][84].

Table 19 shows a comparison of the number of operations performed by the three protocols. Similarly to the protocol of Pazos-Revilla et al. [58] and L. Roman and P. Gondim [82], our protocol does most of the processing work on the entity with enhanced computational features and on the blockchain. On the other hand, EV, RSU, and the pads have fewer computing capacities, and, therefore, conduct less complete operations, which helps reduce the latency of the system.

TABLE 19. COMPARISON OF COMPUTATIONAL COSTS (PROPOSED PROTOCOL PROT_3)

Protocols	**EV**	**CSP BNK/ FS/CCC**	**RSU/ Pad Owner**	**Pad**
Pazos-Revilla et al. [58]	$5Tmp_{exp}$ + $(3+2\psi)Tmp_{hash}$ $+2Tmp_{pair}$	$(8n+3)Tmp_{exp}$ $+(4n+3)Tmp_{mul}$ $+2nTmp_{pair}$	$2Tmp_{pair}$ + $3nTmp_{hash}$	$2n(\psi)Tmp_{hash}$
L. Roman and P. Gondim [82]	$3Tmp_{chaos} + (1+\psi)Tmp_{hash}$	$4nT_{exp} + 6n\,Tmp_{chaos}$ $+2n\,Tmp_{mul}$	$2nT_{hash}$ $+4n\,Tmp_{chaos}$	$n(\psi)T_{hash}$
Proposed Protocol	$\boldsymbol{1Tmp_{chaos}} +$ $\boldsymbol{(7+\psi)Tmp_{hash}}$ $\boldsymbol{+2\,Tmp_{mul}}$	$\boldsymbol{4nTmp_{chaos}} +$ $\boldsymbol{2nTmp_{hash}}$ $\boldsymbol{+n\,Tmp_{mul}}$	$\boldsymbol{5nTmp_{chaos}} +$ $\boldsymbol{nTmp_{hash}}$ $\boldsymbol{+n\,Tmp_{mul}}$	$\boldsymbol{n(\psi)T_{hash}}$

Figure 36 shows a comparison of computational costs in the authentication phase among our protocol and those of Pazos-Revilla et al. [58] and L. Roman and P. Gondim [82] . Our protocol has a better computational cost for EVs, CCC, and RSUs due to the use of chaos-based encryption and Blockchain in the validation of identities and tickets.

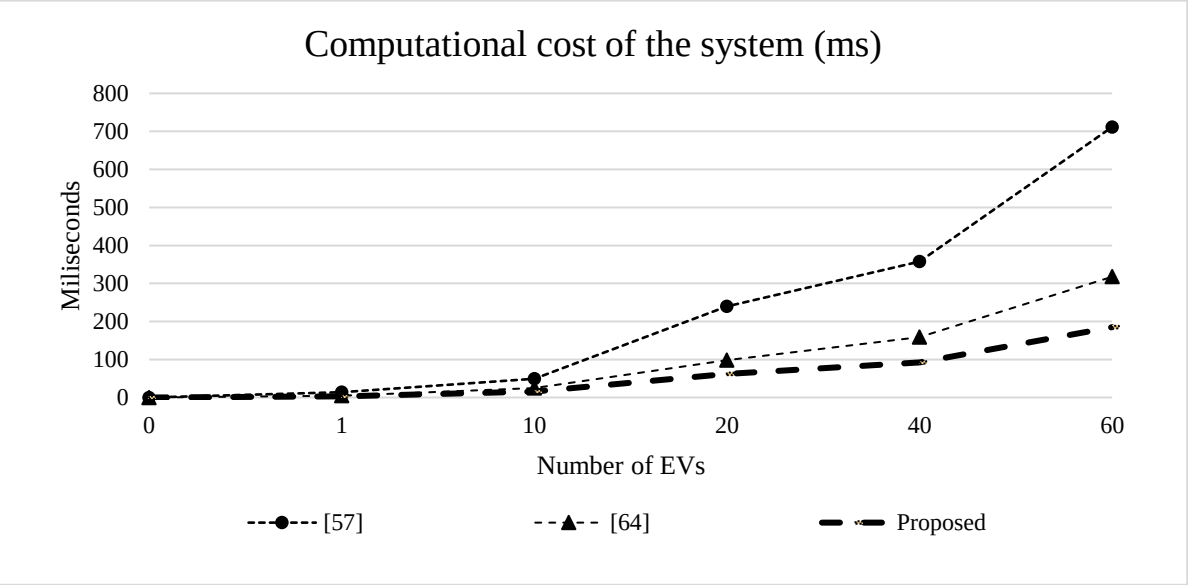


FIGURE 36. COMPUTATIONAL COSTS (PROPOSED PROTOCOL PROT_3)

In what follows is the calculation of the energy costs of the protocols for showing the importance of optimizing authentication protocols towards reducing the amount of energy used in the systems.

5.2.1.3. Energy Costs

The energy costs from the energy consumed in the execution of cryptographic operations in the protocols were compared by Equation $\boldsymbol{E_C} = T_{EX} * W$, where T_{EX} is the execution time and W is the maximum power CPU. W = 10.88 watts [76][77] was assumed for the comparison of the energy costs of our protocol with those of Pazos-Revilla et al. [58] and L. Roman and P. Gondim [82] (see Table 20). According to Figure 37, our protocol consumed the lowest energy.

TABLE 20. COMPARISON OF ENERGY COSTS (PROPOSED PROTOCOL PROT_3).

Protocols	Equation
Pazos-Revilla et al. [58]	$E_{cost} = \left((\boldsymbol{8n+3})\boldsymbol{Tmp_{exp}} + (\boldsymbol{6}+\boldsymbol{4\psi})\boldsymbol{nTmp_{hash}} + (\boldsymbol{2+2n})\boldsymbol{Tmp_{pair}} + (\boldsymbol{4n+3})\boldsymbol{Tmp_{mul}}\right) * 10{,}88W$
L. Roman and P. Gondim [82]	$E_{cost} = \left(\boldsymbol{9} + \left((\boldsymbol{1+2\psi})+\boldsymbol{2}\right)\boldsymbol{nTmp_{hash}} + \boldsymbol{2nTmp_{mul}} + \boldsymbol{4nT_{exp}} + \boldsymbol{13nT_{chaos}}\right) * 10{,}88W$
Proposed Protocol	$E_{cost} = (\boldsymbol{16+2\psi})\boldsymbol{nTmp_{hash}} + \boldsymbol{4nTmp_{mul}} + \boldsymbol{4nT_{chaos}}) * 10{,}88W$

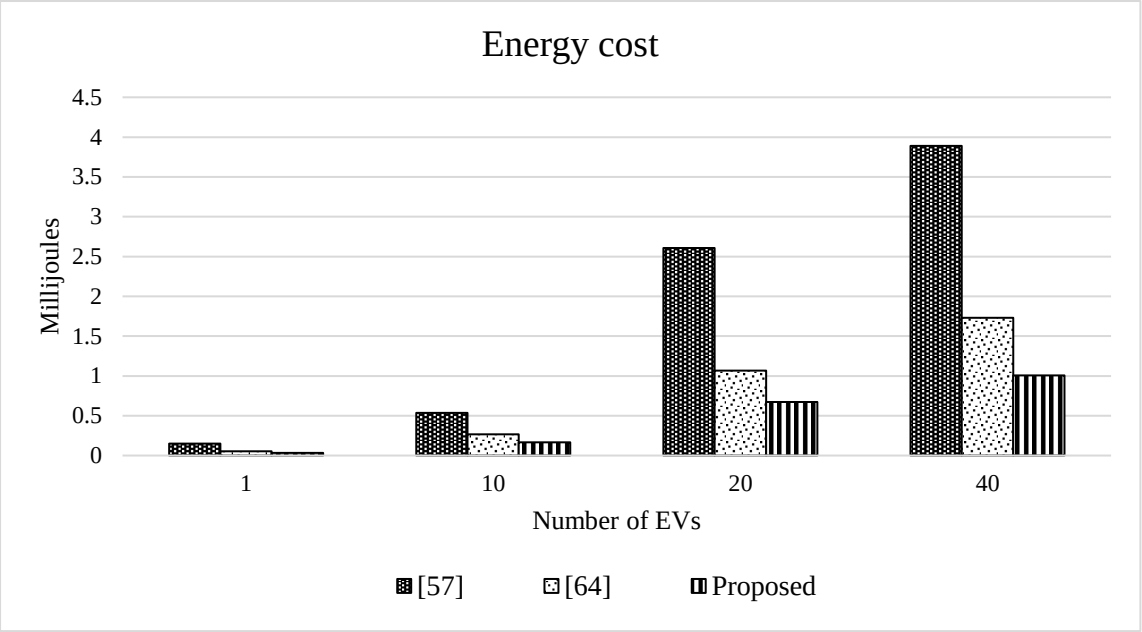


FIGURE 37. COMPARISON OF ENERGY COSTS (PROPOSED PROTOCOL PROT_3)

5.3. Security and Performance Analyses

A security analysis of the protocol and a comparison of its security properties and resistance to cyber attacks with those of other blockchain-based authentication protocols in EV charging systems were conducted.

5.3.1. Security Properties

The security analysis focused on the way the protocol guarantees integrity, privacy, confidentiality, mutual authentication, forward secrecy, and backward secrecy.

- Integrity: the protocol guarantees the content of the messages using a hash function on their sending in the messages exchanged as entities.

- Privacy and Anonymity: During user registration in the system, the TA securely stores the identity and other personal data of the EV owner. Pseudo identities are used for group generation, ticket purchase, and upload processes, thus protecting the user´s real identity.

- Confidentiality: All messages exchanged through non secure channels are encrypted with session keys, guaranteeing unauthorized people or systems cannot obtain the information sent.

- Mutual authentication: In the authentication process, two parties exchanging messages are ensured to authenticate each other in the group creation, ticket purchase, and upload phases using challenges and public keys taken from the blockchain.

- Perfect Forward Secrecy (PFS): The use of random variables for the generation of session keys between the EV and charging station entities guarantees the system's PFS. However, if a new group member is added to the charging station entity group, a new group key is generated without the new member knowing the previous one.

- Perfect Backward Secrecy (PBS): when a member of the group leaves it, the remaining members generate a new group key, preventing the member who has left from decrypting the group's messages after their exit, thus guaranteeing PBS. Regarding EVs, the use of session keys ensures every new communication is encrypted with a different key, preventing an attacker from using an old session key to decrypt new messages.

- Unlinkability: Because pseudo-identities are used for the authentication process and the user's real identity is securely stored in the TA, no system entity or attacker can link activities on the system to that of the real user.

- Double spending: the publication of tickets used in the CWD-WPT charging station in blockchain prevents them from being reused in the system.

5.3.2. Prevention against attacks

In what follows is a description of the way the Proposed Protocol PROT_3 resists possible attacks:

- Privileged insider attack [80]: This attack is one of the most effective, since some systems have neither a security policy, nor internal security mechanisms that detect and resist it. Below are some essential policies to increase the level of internal security:

    - Classification of duties: a company considers a clear specification of the employees´ roles with duties and restrictions important;
    - Limited privileges: accesses by employees and partners must be clearly limited both physically and in the systems, according to the developed roles;
    - Encrypt sensitive data: confidential and essential information for the company must be properly encrypted and stored in a secure place. Therefore, a Backup plan that enables its recovery in case of failure is required;
    - Awareness of security: clear security policies and processes that enable employees and external partners of the company to understand and comply with them through publications and training; and
    - Defense in depth: The implementation of security layers ensures the company's most sensitive processes can have stricter and deeper security compared to other common processes.

- Random number leakage attack [51]: This attack is resisted through the application of the following operations and controls to the Pseudorandom number generator (PRNG) systems:

    - The initial values must be previously concatenated with a timestamp and processed by a collision resistant hash function;
    - use of a collision resistant hash function in the output of the PRNG;
    - the initial values of the PRNG system must be changed at random periods;
    - an intelligent seed is used at the starting points of the PRNG.

- Replay attack: The use of timestamp in messages exchanged in the protocol guarantees protection against Replay attacks.

- Man-in-the-middle attack: Due to the use of chaos-based encryption in a challenge-response scheme and hash functions to ensure message integrity, valid users can detect if an attacker is in the connection.

- DoS attack: due to the benefits of Blockchain, which is a distributed system, DoS attacks can be resisted, since information can be verified in any RSU or FS of the system if any other has met other requests.

- Injection attacks: this attack is resisted through the application of hash functions for the generation of a fingerprint of the messages exchanged between entities; if the message is manipulated during its journey to the destination, the receiver can detect the changes.

- Impersonation attack: The protocol resists impersonation attack using Blockchain, which guarantees the system has fast and truthful knowledge on the public keys and other data necessary for the verification of a user´s validity.

- Known key attack: The protocol resists it using random elements and session keys that prevent an attacker from using an old key to access the system. On the other hand, Blockchain guarantees the system has quick and accurate knowledge on the tickets used, preventing them from being reused.

- Masquerade attack: it is resisted because the entities must use their private keys for generating session keys, thus guaranteeing only the valid entities of the system can successfully authenticate themselves.

- Resistance password-guessing attack: The use of random elements for the generation of session keys and tickets prevents an attacker from guessing the keys or valid tickets that can be used in the system.

Table 21 shows a comparison of the security properties of the proposed protocol with those of other schemes.

TABLE 21. COMPARISON OF SECURITY PROPERTIES (PROPOSED PROTOCOL PROT_3)

	Pazos-Revilla et al. [58]	L. Roman and P. Gondim [82]	Proposed Protocol PROT_3
Integrity	Yes	Yes	Yes
Privacy and Anonymity	Yes	Yes	Yes
Confidentiality	Yes	Yes	Yes
Mutual authentication	Yes	Yes	Yes
PFS	Untreated	Yes	Yes
PBS	Yes	Yes	Yes
Unlinkability	Yes	Yes	Yes
Double spending	Yes	Yes	Yes
Privileged insider attack	Untreated	Yes	Yes
Random number leakage attack	Untreated	Yes	Yes
Replay attack	Yes	Yes	Yes
Man in the middle attack	Yes	Yes	Yes
DoS attack	Untreated	Yes	Yes
Injection attacks	Untreated	Yes	Yes
Impersonation attack	Yes	Yes	Yes
Known key attack	Untreated	Yes	Yes
Masquerade attack	Untreated	Yes	Yes
Resistance password guessing attack	Yes	Yes	Yes

5.3.3.AVISPA Verification

5.3.3.1. Modeling of the Proposed Protocols in HLPSL

The protocols must be modeled according to HLSPL for their evaluation by AVISPA. Figures 38 and 39 show some parts of the EV authentication phase modeling of the proposed protocols Figure 38 displays the modeling of EV and session behavior in HLPSL code.

Role of EV in HLPSL	HLPSL codification of the role session
role role_EV(EV:agent,G:agent,RSU:agent,BC:agent,H1:hash_func,SND,RCV:channel(dy)) played_by EV def= local State:nat,Vev:text,T1:text,T2:text,Yev:public_key,Xrsu:text,Ks:symmetric_key,T3:text,Tao:text,Eta:text,Beta:text,Mul:hash_func,Xev:text,Cheby:hash_func,Yrsu:public_key init State := 0 transition 1. State=0 /\ RCV(start) =|> State':=1 /\ Vev':=new() /\ T1':=new() /\ Yev':=new() /\ SND(H1(T1'.Vev'.Yev').T1'.Vev') 2. State=1 /\ RCV({Eta'}_Ks'.H1(Cheby(Cheby(Mul(Tao'.Xrsu').Beta').Yev)).Tao'.T2'.H1({Eta'}_Ks'.H1(Cheby(Cheby(Mul(Tao'.Xrsu').Beta').Yev)).Tao'.T2')) =|> State':=2 /\ secret(Ks',sec_1,{}) /\ T3':=new() /\ Yrsu':=new() /\ Xev':=new() /\ SND(H1(Eta'.Cheby(Cheby(Mul(Tao'.Xev').Beta').Yrsu').Tao'.T3').T3') end role	role session1(EV:agent,G:agent,RSU:agent,BC:agent,H1:hash_func) def= local SND4,RCV4,SND3,RCV3,SND2,RCV2,SND1,RCV1:channel(dy) composition role_BC(EV,G,RSU,BC,H1,SND4,RCV4) /\ role_G(EV,G,RSU,BC,H1,SND3,RCV3) /\ role_RSU(EV,G,RSU,BC,H1,SND2,RCV2) /\ role_EV(EV,G,RSU,BC,H1,SND1,RCV1) end role

FIGURE 38. EV AND SESSION ROLE IN HLPSL FOR PROPOSED PROTOCOL PROT_3

Figure 39 shows the security objectives of the authentication phase of protocol PROT_3. Namely:

- secrecy_of sec_1: keep secret k_s;
- authentication_on auth_2: EV authenticates RSU on ε;
- authentication_on auth_3: RSU authenticates EV on $H(K_s)$;

Proposed Protocol PROT_3
goal secrecy_of sec_1 authentication_on auth_2 authentication_on auth_3 end goal

FIGURE 39. SECURITY OBJECTIVES AND RELATED SECRETS OF THE PROTOCOL PROT_3 IN HLPSL

5.3.3.2. Security Check Results

Protocol PROT_3 was simulated in AVISPA with the use of CL-AtSe and OFMC backends and the results confirmed its security in the EV authentication phase (Fig. 40).

Backend	CL-AtSe	OFMC
Proposed Protocol PROT_3	SUMMARY SAFE DETAILS BOUNDED_NUMBER_OF_SESSIONS TYPED_MODEL PROTOCOL /home/span/span/testsuite/results/hlpslGenFile.if GOAL As Specified BACKEND CL-AtSe STATISTICS Analysed : 7 states Reachable : 5 states Translation: 0.00 seconds Computation: 0.00 seconds	% OFMC % Version of 2006/02/13 SUMMARY SAFE DETAILS BOUNDED_NUMBER_OF_SESSIONS PROTOCOL /home/span/span/testsuite/results/hlpslGenFile.if GOAL as_specified BACKEND OFMC COMMENTS STATISTICS parseTime: 0.00s searchTime: 0.02s visitedNodes: 19 nodes depth: 6 plies

FIGURE 40. SECURITY SIMULATION RESULTS FOR PROPOSED PROTOCOL PROT_3

5.4. Summary

This chapter described the design and operation of protocol PROT_3 for operation in a decentralized system. The protocol is based on blockchain and chaotic encryption and considers a CWD-WPT charging station with centralized control over a VANET network linked with cloud computing and fog computing. Such an architecture takes advantage of the scalability and high performance and supports the high mobility and low latency of the system. Compared with other protocols, it guarantees the security of information of both users and system with lower computational costs, due to the use of blockchain and chaos-based cryptography, which enables secure authentication with fewer computational operations. The protocol obtained excellent results in terms of security and performance, compared to other schemes.

.

6. TRUST MANAGEMENT AND PROTOCOL DESIGN AND EVALUATION

Various cryptographic techniques are used for authentication protocol design - bilinear pairing, for example, offers several advantages in modeling the protocol; however, it is one of the cryptographic schemes of highest computational cost, compared to elliptic curve cryptography and cryptography based on hash functions [82]. The use of trust management simplifies some processes or eliminates others, decreasing the computational costs generated by encryption[85][86].

Trust models are a security tool that helps to identify malicious elements in systems through feedback (classification) from its participants and evaluation of the reliability of its entities. However, the system faces challenges, of which one is the fact some personal and confidential data of some trust models are exposed to other system entities that should not have such information. Blockchain can be used to overcome it [67] [56].

Blockchain provides VANET networks with transparency in their operation (non-repudiation), resistance to attacks, and fast and efficient validation of user credentials in the authentication process for either authorizing, or denying access to the system [28]. Furthermore, it ensures high service availability due to its decentralized design [30].

This chapter introduces an authentication protocol with a Blockchain-based trust model and bilinear pairing-based cryptography for a CWD-WPT charging system in a cloud and fog computing environment that guarantees privacy and integrity of messages and mutual authentication between EVs and the charging station.

6.1. Network Model and Adversary Model

This section focuses on the network and adversary models considered in this study.

6.1.1.Network Model

A centralized CWD-WPT System supported by a consortium blockchain system has been considered. In the PROT_4, the TA, FS and EVs form the Blockchain network. The blockchain system is supported by TA, where the genesis block is generated and the FSs receive messages and create the blocks. The consensus system used by the FSs is the Redundant Byzantine Fault Tolerance (RBFT) for validate and publicize the creation of a new block to all network participants.

Each block contains the cryptographic hashes of the records, including information about the hash value of the previous block, thus forming a chain of data, i.e. a blockchain.

Blocks are composed of a block header and a block body. The first results from executing a hash function on the header of the previous block, a random number (nonce) and the root of Merkle (binary hash trees). On the other hand, the block body stores the data of the tickets used by the EVs to access the system, the EV's trust assessment and the access token generated in the authentication process between the EV and the CWD-WPT charging station, and other additional blockchain-related information (see fig. 3).

The Merkle root is generated from the "Merkle trees" algorithm that groups all transactions to be registered in the block, the description of how this algorithm works is given in section 2.1.9 (figure 4).

The considered architecture is formed by the following components (Fig. 41):

- Trusted Authority (TA): installed on the cloud, it registers and generates system and user keys and creates the genesis block of the blockchain system;
- A company charging server (CCS), installed on the cloud;
- RSUs (road-side units), implemented by access points deployed at the margins of the road
- Fog servers, installed near the RSUs;
- Charging pads, installed on the floor of the charging station and used for inducing an electric charge to the moving EVs; and
- electrical vehicles (EVs).

Wireless networks (such as 5G) are used for communication among EVs, FS, and RSUs. On the other hand, communications between pads and EVs are supported by short-range wireless networks.

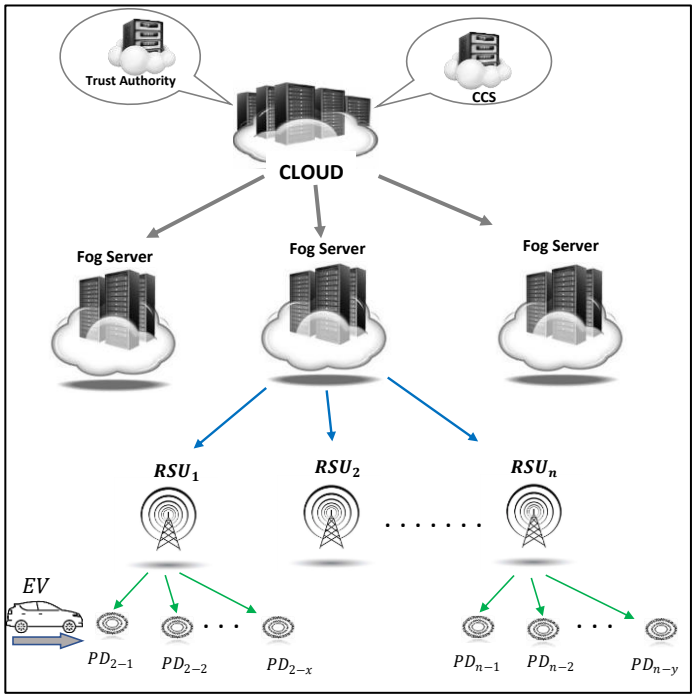


FIGURE 41. NETWORK MODEL (PROPOSED PROTOCOL PROT_4).

6.1.2. Adversary (attack) Model

This study considered the attack model of Dolev-Yao [74] and that an attacker can reproduce messages, unidirectional functions are unbreakable, and information cannot be obtained from encrypted messages if the attacker does not have the key to decipher them.

In the scheme, TA and CCS are trusted, hence, safe for storing EV identification and bank details for ticket purchase. On the other hand, FS, RSU, and pads must not know the identity of the EV or the one of its owner.

6.2. PROT_4 - Trust Management and Authentication Protocol for CWD-WPT Charging Stations

The proposed protocol based on [65] comprises the following four phases (Fig. 42):

- System Initialization, which defines the functions, properties, and keys (private and public) of the system.
- Registration: users are registered in the system and, once registered, the system delivers the keys (public and private) to be used in subsequent authentication processes.
- Ticket Purchase: users purchase multiple tickets to be used at the charging station.
- Charging Request and Authentication: the user requests the charging station service and, upon authentication of the tickets used in the FS, if the ticket is valid, the fs calculates the EV trust, as indicated in section 2.5, which describes the calculation of the global trust value (GTV) (EQ. 40), from explicit (Eq.34) and implicit trusts (Eq.39). IF the system classifies the EV as trusting (GTV≥TH - where TH is the trust threshold), the FS announces the results in the blockchain and proceeds with access control and provision of the charging service for the EV according to protocol PROT_4; otherwise (no-trust EV), the FS will announce the results on the blockchain and proceed to control access and provision of the charging service for the EV according to protocol described in L. Roman and P. Gondim [65].

In the proposed protocol, FS is considered safe, RSUs and pads are considered safe but curious, and EVs are considered unsafe. On the other hand, communications that support the functioning of the system, but are not directly part of the authentication or access control processes, which are the focus of this thesis, are therefore assumed to be secure, that is, communications are secure between:

- the FS and the RSU, on all phases,
- RSUs and pads, on all phases,
- the EV and the CCS, on the EV registration and purchase of tickets phases.

Communications are insecure between:

- the EVs and the FS in the charging request and EV authentication phase,
- the EVs and the RSU in the charging request and EV authentication phases,
- the EVs and pads in the charging request and EV authentication phases.

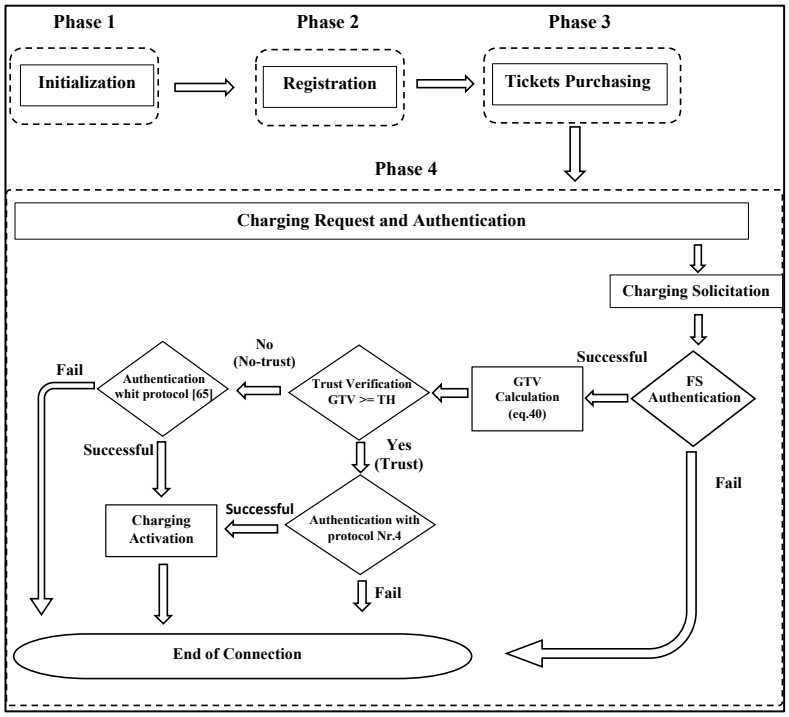


FIGURE 42. PHASES OF THE PROPOSED PROTOCOL PROT_4.

Each FS has a private key x_{fs} and a public key Y_{fs}. RSU also has a private key x_{rsu} , a public key Y_{rsu} , and a group key K_{G-RSU} and is connected to the fog server. On the other hand, the pads are connected to RSUs and a group key K_{G-pads} is defined for the pads.

The phases of the PROT_4 protocol are described in what follows.

1st phase: **System Initialization**

The System boots, as indicated in [65], and the following keys are set:

- CCS defines an elliptical curve on a finite field E (Fq), two sets G_1 (additive) and G_2 (multiplicative), a random value $P \in G_1$ and two collision-free hash functions $H: \{0,1\}^* \rightarrow G_1, H_1: \{0,1\}^*. G \rightarrow \mathbb{Z}_q^*$
- Master private key $X_{ccs}, \in Z_q^*$;
- Global public key $Y_{pub} = X_{ccs} * P$,
- CCS calculates its own pair of public $Q_{ccs} = H(ID_{ccs})$ and private key $S_{ccs} = X_{ccs} * Q_{ccs}$
- Finally the parameters $\{G_1, G_2, \hat{e}, P, H, H_1, Y_{pub}, Q_{ccs}\}$ are published.

2nd phase: **EV registration**

When a user decides to recharge their EV, he/she uses a CWD-WPT station. The first step is to register through a secure channel in CCS. The user must select a private key $x_{ev} \in Z_p^*$ and calculate public key $y_{ev} = x_{ev} * P$. The user and EV data (vehicle charging parameters (VCP) (e.g., battery type, charging level, among others), identity (ID_{ev}), and public key are sent to the CCS for storage. Finally, a certificate $Cert_{ev} = X_{ccs} * Q_{ev}$, where $Q_{ev} = H(ID_{ev})$, is created by the CCS and sent jointly with Q_{ev} to the EV.

3rd phase: **Tickets Purchase**

The ticket purchase process is similar to that of [65], as follows.

The EV sends CCS the first message, m_1, requesting the purchase of n tickets:

$$m_1 = \{n, ID_{EV}, Cert_{EV}\}$$

The CCS receives it and generates n random values $\{r_1, r_2, \dots, r_n\} \in Z_q^*$. For each r_i for $0 \leq i \leq n$, $R_i = r_i * P$ is calculated and a message m_2 containing set R=$\{R_1, R_2, \dots., R_n\}$ is sent to the EV:

$$m_2 = \{R\}$$

The EV creates $\boldsymbol{n}$ random pseudonyms $\{PID_1, PID_2, \dots, PID_i \dots., PID_n\}$ and applies a blind signature to each $\boldsymbol{n}$ PID. It then chooses two random numbers $a, b \in Z_q^*$ and computes the blind pseudonym (B) for every pseudonym PID:

$$B_i = H(PID_i, \hat{e}(bQ_{ccs} + R_i + aP\ , Y_{pub}))+b. \quad (117)$$

The EV sends message m_3 with $B = \{B_1, B_2, \dots., B_i, \dots. B_n\}$ to the CCS to receive the system signature.

$$m_3 = \{B\}$$

The CCS receives the message, signs all blind pseudonyms from set B

$$Bs_i = (B_i * S_{ccs}) + (r_i * Y_{pub}). \quad (118)$$

and sends message m_4 ($Bs = \{Bs_1, Bs_2, \ldots., Bs_n\}$) to the EV, which calculates two values (J and L) for signature verification to obtain the signature of each blind pseudonym set $B = \{B_1, B_2, \ldots., B_i, \ldots. B_n\}$.

4th phase: **Charging Request and Authentication**

This phase describes the process of charging request, authentication, verification, and creation of session keys between the CWD-WPT charging station and the EV.

Access to the charging station

When an EV owner wants to recharge the vehicle at a CWD-WPT charging station and has a valid ticket (J, L), the EV chooses a random number $\sigma_{ev} \in Z_n^*$, calculates $\gamma_{ev} = \sigma_{ev} * P$, and sends an m_1 message to the FS

$$m_1 = \{\gamma_{ev}, ts_1, \{(PID_1, J, L), cert_{ev}\}_{y_{fs}}, HMAC(\gamma_{ev}||ts_1||cert_{ev}||Q_{ev})\},$$

where ts_1 is a timestamp.

If the FS verifies the validity of the certificate and the hash of the message to authenticate the EV_i, it checks the validity of the Ticket:

$$L = H(PID_1, \hat{e}(J, P)\hat{e}(Q_{ccs}, Y_{ccs})^{-L}). \quad (119)$$

If the ticket is valid, the system starts validating EV_i trust.

Trust is checked by off-chain evaluations and by running the formulas in Section 2.5. If the check is negative or the trust is below the threshold, the FS sends a message to the RSU to continue with the authentication process described in the protocol proposed in [65]. Otherwise, it sends an m_2 message with an authentication token $TK = x_{fs} * \gamma_{ev}$, random value α_1 (to calculate the hash chain to authenticate with the pads), and γ_{ev} to all system entities for a fast authentication.

$$m_2 = \{TK, \gamma_{ev}, \alpha_1\},$$

Additionally, it sends an m_3 message containing the Token hash, the ticket, and the trusted classification to the Blockchain.

$$m_3 = \{H(TK), H(J, L), TV_{i,j}, H(H(TK), H(J, L), TV_{i,j})\ \},$$

When the messages arrive on the blockchain (in this case, the FSs), they form a body block grouping all transactions (according to m_3) of all the EVs that are using the system. In addition, the FSs create the header of a new blockchain block that contains the Merkle root of the transaction group (m_3), a random number, and the hash of the previous block header. Finally, the first FS that creates a valid block sends the new block for validation to the other FSs, and a new block is then added to the Blockchain, through the use of RBFT consensus algorithm.

It also sends the EV the token and the group key encrypted with the token (*TK*):

$$m_4 = \{\{\alpha_1\}_{TK}, VK, ts_4\},$$

The EV then calculates token $TK = \sigma_{ev} * Y_{fs}$ and the hash of token $VK = H(TK)$ and verifies if $VK =? VK'$. If the verification agrees, the EV decrypts the message and uses α_1 to calculate the hash chain for the authentication with PADs $k_{PH_\psi} = H^{\psi}(\alpha_1)$

$$m_5 = \{H^{\psi}(\alpha_1), ts_6\},$$

Towards authenticating with the following RSU, the EV sends the token and the variable γ_{ev} together with an HMAC that contains the token, the authentication variable, and the applied hash τ (number of RSUs of the CWD-WPT station) in seed α_1.

$$m_6 = \{\gamma_{ev}, HMAC\ (TK, \gamma_{ev}, H^{\tau}(\alpha_1))\},$$

The RSU verifies the HMAC with the values associated with γ_{ev}, sends an encrypted α_{τ} with TK to the EV

$$m_7 = \{\alpha_{\tau}\}_{TK},$$

and an m_8 to the pads with $k_{PH_{\psi}}^{\tau} = H^{\psi}(\alpha_{\tau})$

$$m_8 = \left\{k_{PH_{\psi}}^{\tau}\right\}_{k_{G-pads}}$$

The EV decrypts m_9 and uses α_{τ} to calculate the hash chain and authenticate with the PADs of the following RSUs $k_{PH_{\psi}}^{\tau} = H^{\psi}(\alpha_1)$

$$m_9 = \{k_{PH_{\psi}}^{\tau}\},$$

Finally, the EV_i sends a message with the evaluation of the service from the CWD-WPT station to the off-chain via a secure channel.

$$m_{10+\tau*\psi} = \{TV_{i,j}^{h+1}\}$$

6.3. Comparative Performance Evaluation

A performance analysis of the communication, computational, and energy costs of the protocol was conducted and a comparison with the protocol of [65] was performed. For comparison purposes, a charging station (as described in [65]) was considered with the following characteristics: n EVs ,$\tau = 7$ RSUs, and $\psi = 750$ pads/RSU.

6.3.1. Communication Costs

The communication costs involved the size of the transmitted messages (in bytes) in the "Charging Request and Authentication" phase, but not the headers or control bits inherent to the communication protocol used.

Tables 22 and 23 show the values in bits of the variables used in the protocol and a comparison of its costs, respectively.

TABLE 22. SYMBOLS AND COSTS IN BYTES ([65]) (PROPOSED PROTOCOL PROT_4).

Symbol	Description	Length (Bytes)
ID	Identification	128
PID	Pseudo identity	32
$H(\ )$	Hash function	32
X	Private key	32
Y, Q	Public key	32
k	Session key	32
σ	Digital signature	32
(J, L)	Blind signature	96
ϕ	Pre key of session	32
τ	Number of RSUs per fog server	8
ψ	Number of pads per RSU	8
α, v	Seed	20
t	Timestamp	8
VK	Verification key	32
hash chain request	Hash chain request	8
*	Multiplication operator	-
$\hat{e}$	Bilinear Pairing	-
CCS	Company Charging Server	-
RSU	Roadside unit	-
HMAC	Hash-based message authentication code	32
P	Generator point of the elliptical curve	32

TABLE 23. COMPARISON OF COMMUNICATION COSTS IN BYTES (PROPOSED PROTOCOL PROT_4).

Message	L. Roman and P. Gondim [65]	Proposed Protocol PROT_4
M1	$72n$	$200n$
M2	$104n$	$104n$
M3	$136n$	$104n$
M4	$64n$	$60n$
M5	80n	$32(n * \psi)$
M6	$80(n * \tau)$	$32(n * (\tau - 1))$
M7	$104(n * \tau)$	$32(n * (\tau - 1))$
M8	$16(n * \tau)$	$32n$
M9	$32(n * \tau)$	$32(n * (\tau - 1) * \psi)$
M10	$32n$	$8n$
M11	$32(n * \tau * \psi)$	--
M12	--	--
Total	$\boldsymbol{n\ (488 + 232\tau\ +\ 32\tau\psi)}$	$\boldsymbol{n\ (444 + 64\tau +\ 32\tau\psi)}$

Figure 43 displays the communication costs of the proposed protocol compared with that of L. Roman and P. Gondim [65]. The costs between the two protocols are similar, because the same cryptographic technique is used (hash chain) and the EV and the group of pads of the charging station exchange a larger number of messages (5250 messages) compared to communication between the EV and other entities (22 messages).

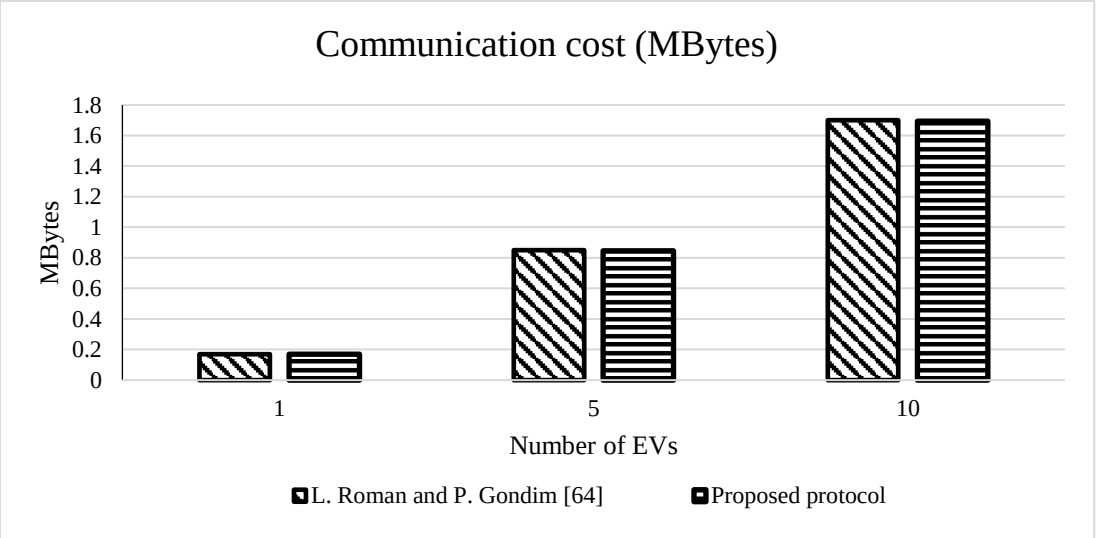


FIGURE 43. COMPARISON OF COMMUNICATION COSTS (PROPOSED PROTOCOL PROT_4).

6.3.2.Computational Costs

The time required for the undertaking of a unit operation and estimated according to the processing power of each entity is considered for the calculation of computational costs. Execution times are based on experiments conducted on computational platforms [82]. In this sense, 3 types of hardware were considered so that the execution times of each unit operation could be obtained. For our study, such hardware types correspond to EV/pad, RSUs, and FS, respectively:

- mobile equipment (processor Qualcomm (R) Octa core 1.5 GHz, 2G RAM).
- a Desktop (Intel (R) Dual core processor 3.1 GHz, 4GB RAM), and
- a Server (Intel (R) Hexa core processor 1.6 GHz and 16G RAM).

A methodology consisting in multiplying the execution time of the function by the number of times the function is executed by each entity calculated the computational costs. Table 24 shows the execution times of the operations used by the protocol.

TABLE 24. COSTS IN ms OF EACH OPERATION AND ENTITY CONSIDERED ([82]) (PROPOSED PROTOCOL PROT_4).

Entity	Costs (ms)						
	Tmp_{mul}	Tmp_{exp}	Tmp_{pair}	Tmp_{hash}	Tmp_{g-ptns}	Tmp_{v-ptns}	Tmp_{Chaos}
EV/Pad	0.29	0.5	0.75	$0.3x10^3$	0.03	0.021	0.95
RSUs	0.22	0.38	0.57	$0.2x10^3$	0.011	0.015	0.07
FS/ CSP	0.17	0.31	0.46	$0.1x10^3$	0.009	0.01	0.05

Table 25 shows a comparison between the protocol of [65] and the proposed one, which was carefully designed to performing more complex operations in devices with greater computational capacity.

TABLE 25.COMPARISON OF COMPUTATIONAL COSTS (PROPOSED PROTOCOL PROT_4)

Protocols	EV	CSP BNK/ CMC /FS	RSU	Pad
L. Roman and P. Gondim [65]	$2T_{mul}$ + $((1+\psi)+4)\tau T_{hash}$ + $1T_{v-sig}$	$2nT_{mul}$+ $1nT_{exp}$ + $1nT_{g-sig}$ +$4nT_{hash}$ + $2nT_{pair}$	$4nT_{hash}$	$n(\psi)T_{hash}$
Protocol PROT_4	$(1+(2+\psi)*\tau)\,T_{hash}$ + $4T_{mul}$	nT_{mul}+ $1nT_{exp}$ +$4nT_{hash}$ + $2nT_{pair}$	$2nT_{hash}$	$n(\psi)T_{hash}$

Figure 44 shows a comparison between the protocol of [65] and the proposed one regarding the different entities considered. With the implementation of computational trust and blokchain, the computational costs of the proposed protocol were better due to the less complex authentication process. Regarding pads, the two protocols show relatively the same cost, since the same cryptographic scheme, i.e., hash chain is used.

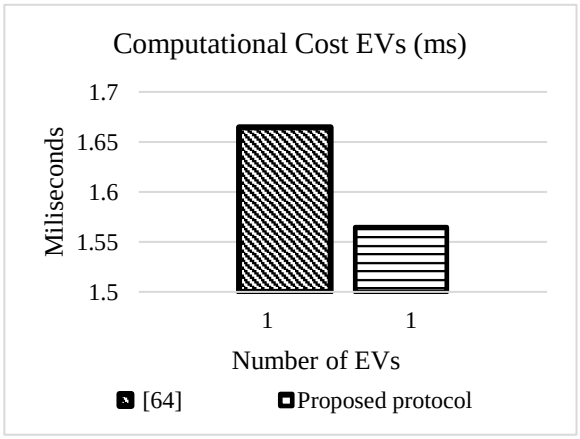


Figure 44.a Computational Costs of EVs

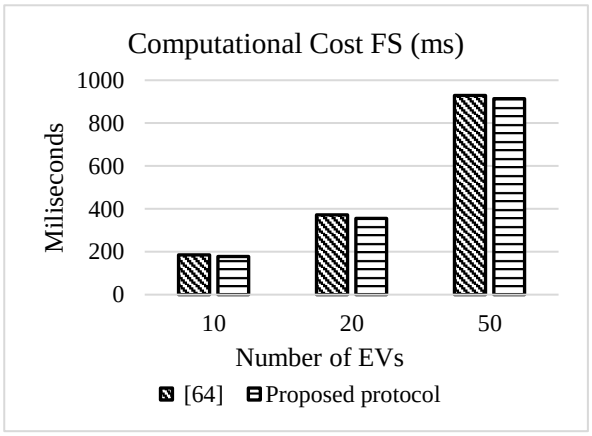


Figure 44.b Computational Costs of FS

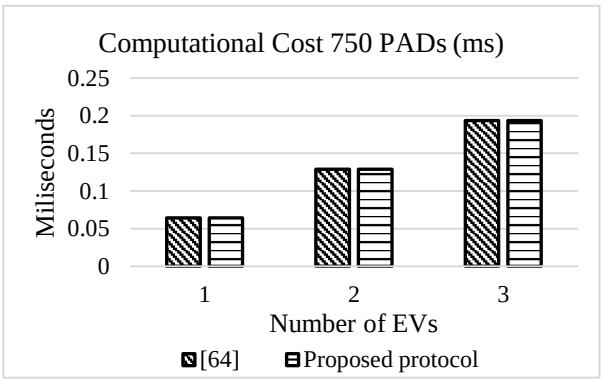


Figure 44.c Computational Costs of pads

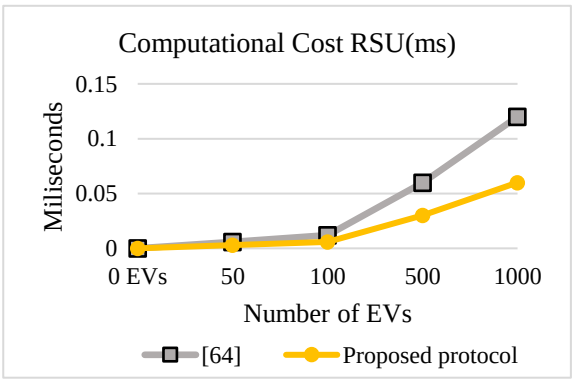


Figure.44.d Computational Costs of RSUs

FIGURE 44. COMPARISON OF COMPUTATIONAL COSTS (PROPOSED PROTOCOL PROT_4).

6.3.3. Energy Costs

This section addresses a comparison of the energy costs of each system′s entity of the proposed protocol and that of the protocol of [65]. The equation used is $\boldsymbol{E_C} = T_{EX} * W$, where T_{EX} is the execution time in ms and W (10.88 watts) is the maximum CPU power – the result is optimal in millijoules. Figure 45 displays differences in energy costs, demonstrating the implementation of computational reliability in the system reduces the energy used.

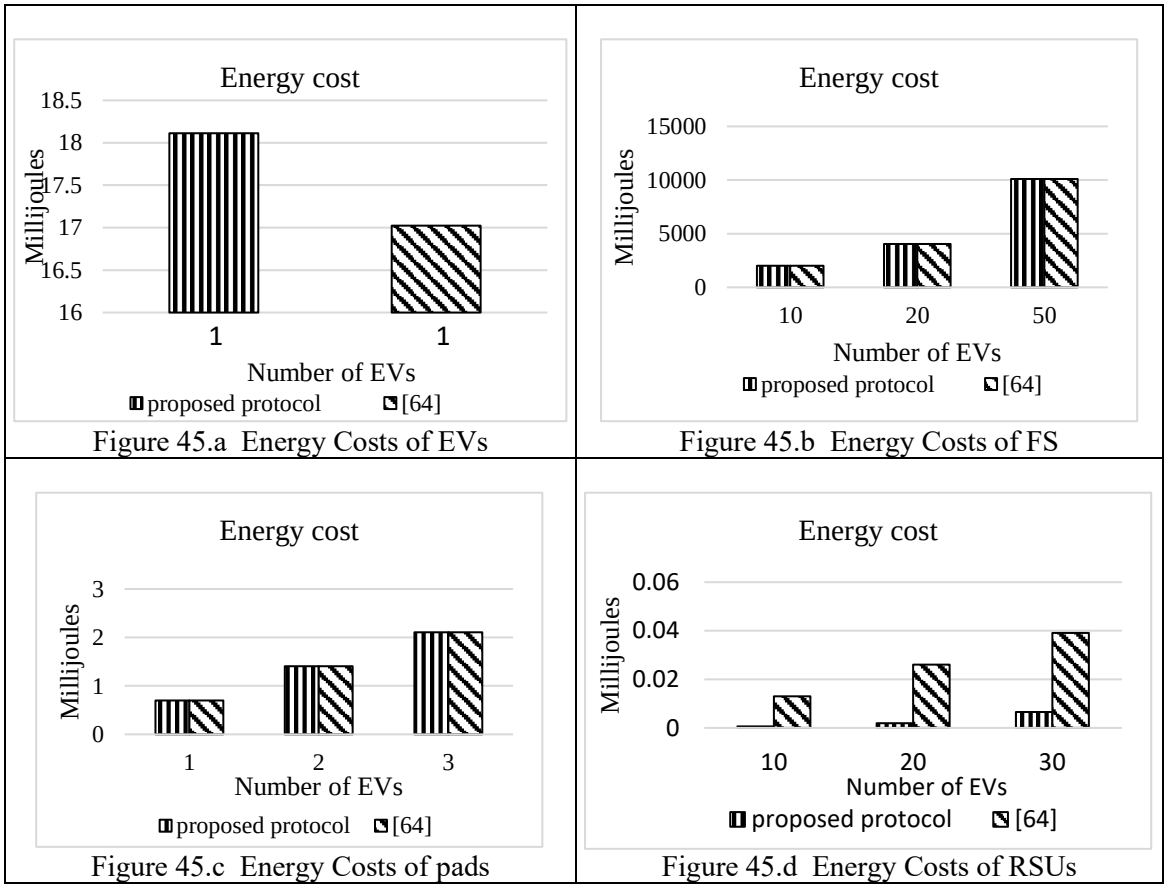


Figure 45.a Energy Costs of EVs

Figure 45.b Energy Costs of FS

Figure 45.c Energy Costs of pads

Figure 45.d Energy Costs of RSUs

FIGURE 45. COMPARISON OF ENERGY COSTS (PROPOSED PROTOCOL PROT_4).

6.4. Security and Performance Analyses

This section is devoted to an analytical description of the security attributes such as mutual authentication, perfect forward secrecy, integrity, and privacy of the proposed protocol, and protection against attacks.

6.4.1. Security Analysis

A security analysis conducted considered the trust of the system, towards improving security and authentication efficiency, since the access control complexity can be adjusted in function of the

EV's trust rating. On the other hand, the communication between a user classified as “trustworthy” and the CWD-WPT recharge system may be considered insecure; therefore, security properties must be guaranteed and attacks must be resisted..

6.4.1.1. Security Properties

This section reports on analysis of the security attributes guaranteed by the proposed protocol.

1) Preservation of privacy: during the ticket purchase process, the identity of the EV user is kept confidential by both CCS and FS and tickets are generated from random elements unrelated to the user. In the system access phase (phase 4), RSUs and pads cannot obtain the user's identity from the token given to the user for authentication.
2) Mutual Authentication: the protocol guarantees double authentication among the EV, FS, RSUs, and pads. The FS authenticates the EV by checking the *HMAC* of message m_1 and the EV authenticates the FS by checking the *VK* of message m_4. The first RSU authenticates the EV against the confirmation of the FS and the EV authenticates the RSU against the *VK* of m_4. The other RSUs authenticate to the EV by verifying the *HMAC* of m_6 and the EV authenticates the other RSUs decrypting with the token (*TK*) and verifying the α_τ of m_7.
3) Integrity protection: integrity is guaranteed with the use of hash functions and HMACs, so that entities receiving a message can verify whether an adversary has altered its content.
4) Perfect Forward Secrecy (PFS): the protocol guarantees PSF through the use of random elements for the generation of session keys. Even if session keys are compromised, messages from other sessions cannot be decrypted.
5) Perfect Backward Secrecy (PBS): this property is guaranteed with the use of session keys – a different key is used for each new communication, preventing an attacker from using an old session key to decrypt new messages.

6.4.1.2. Prevention against attacks

This subsection provides an analysis of the attacks that might affect VANET networks and are resisted by the proposed protocol.

- Impersonation: the system checks whether a ticket is valid and, in case of an invalid one, removes it. Another level of protection is achieved with the use of session keys generated with random elements, which prevent an attacker from using old keys to generate a new key to access the system.
- MitM: hash functions, HMACs, and session keys ensure messages cannot be intercepted, read, and modified by an attacker.
- Replay and Injection: replay attacks are avoided by timestamps and random elements and injection ones are prevented through the use of hash and HMAC functions so that the receiver of the message can validate its integrity.
- Known key: unique tickets for access to the system and the record of their use in the blockchain avoid such an access with a same ticket. On the other hand, session keys are unique and generated with the use of a valid ticket, which prevents their reuse for accessing the system.
- DoS: the protocol can resist the attack at layer 2 and 3 of the TCP/IP model. The FS resists it during the validation of the ticket and the user's trust score. However, depending on the authentication method (standard or trust-based one), RSUs can validate the connections by checking the HMACs of the messages, thus efficiently rejecting not valid connections.
- Masquerade: it is resisted through the use of session keys - an attacker would not be able to represent a valid system message, since random values are used for the obtaining of session keys.

- Unlinkability: the EV identity cannot be linked to the ticket or authentication token by RSU and pads, since it is validated with the use of public system variables.
- Double Spending: the use of a ticket is published on the blockchain by the system, so that the ticket cannot be reused.
- Resistance password-guessing: the protocol resists the attack because tickets and session keys are generated with random elements, which prevents an attacker from guessing the keys required for the use of the recharge system.
- Random number leakage: as suggested in [51], several controls and operations in PRNG are implemented for preventing the attack.
- Privileged insider: such an attack can be avoided with the implementation of a set of security policies that transversally cover all processes that describe the operations towards offering the CWD-WPT loading service. [80] defined the most important policies that minimize risks of the attack.

Table 26 shows a comparison of the security properties of the proposed protocol and those of other schemes for authentication for CWD-WPT load stations.

Table 26. Comparison of security properties (Proposed Protocol PROT_4).

Security Property	[65]	**Proposed Protocol**
Mutual authentication	Yes	Yes
key agreement	Yes	Yes
Confidentiality	Yes	Yes
Integrity	Yes	Yes
Privacy	Yes	Yes
Injection attacks	Yes	Yes
Perfect forward secrecy	Yes	Yes
Perfect backward Secrecy	Untreated	Yes
Replay attack	Yes	Yes
Known key attack	Yes	Yes
DoS attack	Yes	Yes
Man-in-the-middle attack	Yes	Yes
Masquerade attack	Yes	Yes
Impersonation attack	Yes	Yes
Unlinkability	Yes	Yes
Double spending	Yes	Yes
Resistance password-guessing attack	Yes	Yes
Random number leakage attack	Yes	Yes
Privileged insider attack	Yes	Yes

6.5. Summary

A new authentication and access control protocol based on computational trust, blockchain and bilinear pairing encryption for a CWD-WPT system has been designed to improve authentication times and minimize communication, computing, and system energy costs. The protocol uses Blockchain-based computational trust to validate the way to authenticate in the system. If the user's trust is above a certain level, the authentication process in the system is lighter and faster, without neglecting the security of communications. A comparison of the protocol, which includes the trust system, with a similar one that does not include trust, showed its better performance, achieved due to simpler session key generation processes in function of the trust system. Regarding security and performance, the protocol provided excellent results and, therefore, can be a good choice in comparison to other authentication and access control schemes for CWD-WPT systems.

7. CONCLUSIONS

Cyber physical systems (CPS) have shown an evolution due to advances in new technologies such as IoT, characterized by the use of heterogeneous data and knowledge integration. Such an evolution has driven a 4th industrial revolution called Industry 4.0, in which new requirements (e.g., QoS, data volumes, mobility, and interconnection between different devices) have become challenges that must be overcome.

The combination of cloud computing and fog computing in a hierarchical scheme is an effective solution to support next generation VANETs that are compatible with high mobility, low latency, real time services, and connectivity.

Part of the research on EVs has been directed to the creation of VANET networks in a cloud environment for supporting CWD-WTP charging stations. Such stations aim at optimizing and simplifying the charge of EV batteries, since, in this system, cables are not necessary, and power is induced while the EV owners drive to their destination.

This thesis addresses the problems of network security and access control in cloud-based vehicular networks, meeting the most important security requirements, namely authentication, data integrity, confidentiality, access control, non-repudiation, and availability. It aims to contribute to the optimization and security of vehicular networks that support EVs, which has become a trend in several countries due to the global objective of air pollution reduction.

The manuscript introduced four new authentication protocols for two different architectures of a CWD-WPT charging system on a VANET network based on cloud computing and fog, and a trust management scheme based on blockchain represents a current focus of research and development for evaluation in terms of security properties, validation and performance evaluation.

The first architecture is comprised of a centralized system in which a company charging server (CCS) is installed in the cloud computing and a group of secondary servers (fog servers FS) in the fog computing so that each FS groups several RSUs that control a group of pads. Two protocols – one based on bilinear pairing and another, which is a variant of the first, is based on chaotic cryptography - were proposed.

The first scheme introduced a new authentication protocol for CWD-WPT charging systems on a VANET network in a cloud and fog computing environment; it considers a centralized architecture and is based on digital signatures, HMACs and hash chains. Compared to other proposals, it yielded better computational costs and provided better results regarding security analysis.

The second protocol also considers a centralized architecture and uses new cryptographic primitives based on chaotic maps, which have low computational cost compared to cryptographic primitives based on bilinear matching. A comparison with other protocols revealed our scheme enables better handling of security properties and requires lower computational costs, due to the use of chaotic cryptography.

The third protocol uses a cryptographic scheme based on blockchain and chaotic maps to guarantee authentication with low computational cost, protect the anonymity of users, and provide privacy, availability, and integrity to the system. Compared with other protocols, it assures security of information for both users and the system with lower computational costs, due to the use of blockchain.

The three protocols were formally validated by AVISPA tool, which confirmed they are safe against various attacks, including replay, man-in-the-middle, impersonation, and privileged insider ones.

The second architecture involves a decentralized system, in which the Control Charging Center (CCC) is located on the side of the road and connected directly to the pads. The RSUs, FS and a TA are elements that enable the EV to securely communicate with the CCC to perform charging. A protocol based on both blockchain and chaotic cryptography was proposed for the architecture.

The fourth protocol comprised of a centralized system scheme according to which the trust system, the blockchain, and the CWD-WPT charging system are managed in the traditional cloud, whereas fog computing manages the RSUs of the charging stations. The scheme has been adopted by several researchers due to its advantages such as connectivity with low latency and high mobility.

The fourth protocol uses computational trust to validate the way to authenticate in the system. If the user's trust is above a certain level, the authentication process in the system is lighter and faster, without neglecting the security of communications. A comparison of the protocol, which includes the trust system, with a similar one that does not include trust, showed its better performance, achieved due to simpler session key generation processes in function of the trust system.

Moreover, it is expected the production of a trust management scheme based on blockchain, accompanied by the submission and review of at least one paper.

Current work involves the concepts of chaos and blockchain cryptography, in addition to the creation and submission of articles produced from the second and third protocols. Additionally, an extension and technical deepening on the foundations and applications of chaos-based cryptosystems and blockchain to the improvement of information security must be accomplished.

Future work will involve security protocols and mechanisms for the integration of CWD-WPT systems with 5G and 6G communication networks, as well as performance evaluation studies of CWD-WPT charging systems based on discrete-event simulation.

BIBLIOGRAPHIC REFERENCES

[1] M. Karaköse and H. Yetiş, "A cyberphysical system based mass-customization approach with integration of industry 4.0 and smart city," *Wireless Communications and Mobile Computing*, vol. 2017, 2017, doi: 10.1155/2017/1058081.

[2] H. I. Al-Salman and M. H. Salih, "A review Cyber of Industry 4.0 (Cyber-Physical Systems (CPS), the Internet of Things (IoT) and the Internet of Services (IoS)): Components, and Security Challenges.," *Journal of Physics: Conference Series*, vol. 1424, no. 1, 2019, doi: 10.1088/1742-6596/1424/1/012029.

[3] A. Khalid, P. Kirisci, Z. H. Khan, Z. Ghrairi, K. D. Thoben, and J. Pannek, "Security framework for industrial collaborative robotic cyber-physical systems," *Computers in Industry*, vol. 97, pp. 132–145, 2018, doi: 10.1016/j.compind.2018.02.009.

[4] J. Sathishkumar and D. R. Patel, "Enhanced location privacy algorithm for wireless sensor network in Internet of Things," *2016 International Conference on Internet of Things and Applications, IOTA 2016*, pp. 208–212, 2016, doi: 10.1109/IOTA.2016.7562723.

[5] T. Park, N. Abuzainab, and W. Saad, "Learning How to Communicate in the Internet of Things: Finite Resources and Heterogeneity," *IEEE Access*, vol. 4, pp. 7063–7073, 2016, doi: 10.1109/ACCESS.2016.2615643.

[6] T. R. Raddo, R. Nobrega, S. Rommel, B. Cimoli, A. L. Sanches, and I. T. Monroy, "6G and Fog Node Mobile Systems for Cooperative, Autonomous, and Dynamic Applications," *2019 SBMO/IEEE MTT-S International Microwave and Optoelectronics Conference, IMOC 2019*, pp. 2019–2021, 2019, doi: 10.1109/IMOC43827.2019.9317640.

[7] F. J. Soares, D. Rua, C. Gouveia, B. D. Tavares, A. M. Coelho, and J. A. P. Lopes, "Electric Vehicles Charging: Management and Control Strategies," *IEEE Vehicular Technology Magazine*, vol. 13, no. 1, pp. 130–139, Mar. 2018, doi: 10.1109/MVT.2017.2781538.

[8] X. Mou, O. Groling, and H. Sun, "Energy-Efficient and Adaptive Design for Wireless Power Transfer in Electric Vehicles," *IEEE Transactions on Industrial Electronics*, vol. 64, no. 9, pp. 7250–7260, 2017, doi: 10.1109/TIE.2017.2686299.

[9] T. V. Theodoropoulos, I. G. Damousis, and A. J. Amditis, "Demand-Side Management ICT for Dynamic Wireless EV Charging," *IEEE Transactions on Industrial Electronics*, vol. 63, no. 10, pp. 6623–6630, 2016, doi: 10.1109/TIE.2016.2570198.

[10] Y. J. Jang, "Survey of the operation and system study on wireless charging electric vehicle systems," *Transportation Research Part C: Emerging Technologies*, no. November 2017, pp. 0–1, 2018, doi: 10.1016/j.trc.2018.04.006.

[11] A. G. Batres, A. Moghe, and J. Taiber, "A communication architecture for wireless power transfer services based on DSRC technology," *2016 IEEE Transportation Electrification Conference and Expo, ITEC 2016*, 2016, doi: 10.1109/ITEC.2016.7520269.

[12] K. Lee, Z. Pantic, and S. M. Lukic, "Reflexive field containment in dynamic inductive power transfer systems," *IEEE Transactions on Power Electronics*, vol. 29, no. 9, pp. 4592–4602, 2014, doi: 10.1109/TPEL.2013.2287262.

[13] D. Bavastro *et al.*, "Design of wireless power transmission for a charge while driving system," *IEEE Transactions on Magnetics*, vol. 50, no. 2, pp. 2–5, 2014, doi: 10.1109/TMAG.2013.2283339.

[14] C. Li, T. Ding, X. Liu, and C. Huang, "An Electric Vehicle Routing Optimization Model with Hybrid Plug-In and Wireless Charging Systems," *IEEE Access*, vol. 6, pp. 27569–27578, 2018, doi: 10.1109/ACCESS.2018.2832187.

[15] K. Nahrstedt, H. Li, P. Nguyen, S. Chang, and L. Vu, "Internet of mobile things: Mobility-driven challenges, designs and implementations," *Proceedings - 2016 IEEE 1st International Conference on Internet-of-Things Design and Implementation, IoTDI 2016*, pp. 25–36, 2016, doi: 10.1109/IoTDI.2015.41.

[16] R. Shrestha, R. Bajracharya, and S. Y. Nam, “Challenges of Future VANET and Cloud-Based Approaches,” *Wireless Communications and Mobile Computing*, vol. 2018, 2018, doi: 10.1155/2018/5603518.

[17] M. Ziqian, Z. Guan, Z. Wu, A. Li, and Z. Chen, “Security Enhanced Internet of Vehicles with Cloud-Fog-Dew Computing,” *Zte Communications*, vol. 15, no. S2, pp. 47–51, 2018, doi: 10.3969/j.issn.1673-5188.2017.S2.008.

[18] Q. G. K. Safi, S. Luo, C. Wei, L. Pan, and Q. Chen, “PIaaS: Cloud-oriented secure and privacy-conscious parking information as a service using VANETs,” *Computer Networks*, vol. 124, pp. 33–45, 2017, doi: 10.1016/j.comnet.2017.06.001.

[19] C. Huang, R. Lu, and K. K. R. Choo, “Vehicular Fog Computing: Architecture, Use Case, and Security and Forensic Challenges,” *IEEE Communications Magazine*, vol. 55, no. 11, pp. 105–111, 2017, doi: 10.1109/MCOM.2017.1700322.

[20] A. Singh and K. Chatterjee, “Cloud security issues and challenges: A survey,” *Journal of Network and Computer Applications*, vol. 79, no. August 2016, pp. 88–115, 2017, doi: 10.1016/j.jnca.2016.11.027.

[21] R. Akalu, “Privacy, consent and vehicular ad hoc networks (VANETs),” *Computer Law and Security Review*, vol. 34, no. 1, pp. 175–179, 2018, doi: 10.1016/j.clsr.2017.06.006.

[22] R. G. Engoulou, M. Bellaïche, S. Pierre, and A. Quintero, “VANET security surveys,” *Computer Communications*, vol. 44, pp. 1–13, 2014, doi: 10.1016/j.comcom.2014.02.020.

[23] R. Hussain, D. Kim, M. Nogueira, J. Son, A. Tokuta, and H. Oh, “A New Privacy-Aware Mutual Authentication Mechanism for Charging-on-the-Move in Online Electric Vehicles,” *Proceedings - 11th International Conference on Mobile Ad-Hoc and Sensor Networks, MSN 2015*, pp. 108–115, 2016, doi: 10.1109/MSN.2015.31.

[24] S. Gunukula, A. B. T. Sherif, M. Pazos-Revilla, B. Ausby, M. Mahmoud, and X. S. Shen, “Efficient scheme for secure and privacy-preserving electric vehicle dynamic charging system,” *IEEE International Conference on Communications*, 2017, doi: 10.1109/ICC.2017.7997252.

[25] K. Rabieh and M. Wei, “Efficient and privacy-aware authentication scheme for EVs pre-paid wireless charging services,” *Communication and Information Systems Security Symposium*, 2017, doi: 10.1109/ICC.2017.7996868.

[26] M. Luo, Y. Zhang, M. K. Khan, and D. He, “An efficient chaos-based 2-party key agreement protocol with provable security,” *International Journal of Communication Systems*, vol. 30, no. 14, pp. 1–9, 2017, doi: 10.1002/dac.3288.

[27] D. Abbasinezhad-Mood, A. Ostad-Sharif, and M. Nikooghadam, “Novel chaotic map-based privacy-preserving authenticated key agreement scheme without the electricity service provider involvement,” *Security and Privacy*, vol. 2, no. 5, pp. 1–17, 2019, doi: 10.1002/spy2.74.

[28] J. Grover, “Security of Vehicular Ad Hoc Networks using blockchain: A comprehensive review,” *Vehicular Communications*, vol. 34, p. 100458, 2022, doi: 10.1016/j.vehcom.2022.100458.

[29] S. Abbas, M. A. Talib, A. Ahmed, F. Khan, S. Ahmad, and D. H. Kim, “Blockchain-based authentication in internet of vehicles: A survey,” *Sensors*, vol. 21, no. 23, 2021, doi: 10.3390/s21237927.

[30] Q. Feng, D. He, S. Zeadally, and K. Liang, “BPAS: Blockchain-Assisted Privacy-Preserving Authentication System for Vehicular Ad Hoc Networks,” *IEEE Transactions on Industrial Informatics*, vol. 16, no. 6, pp. 4146–4155, 2020, doi: 10.1109/TII.2019.2948053.

[31] A. Menezes, “An introduction to pairing-based cryptography,” *Math Subject Classification Primary 94A60*, pp. 47–65, 2012, doi: 10.1090/conm/477/09303.

[32] R. Dutta, R. Barua, and S. Palash, “Pairing-Based Cryptographic Protocols : A Survey,” *IACR Cryptology ePrint Archive*, pp. 1–45, 2004.

[33] D. Boneh, B. Lynn, and H. Shacham, “Short Signatures from the Weil Pairing,” *Boyd, C (eds) Advances in Cryptology — ASIACRYPT 2001*, pp. 514–532, 2001, doi: 10.1007/3-540-45682-1_30.

[34] F. Zhang and K. Kim, “ID-Based Blind Signature and Ring Signature from Pairings,” in *Advances in Cryptology — ASIACRYPT 2002*, 2002, vol. 2501, no. December 2002, pp. 533–547, doi: 10.1007/3-540-48910-X.

[35] Y. Hu, M. Jakobsson, and A. Perrig, “Efficient Constructions for One-way Hash Chains,” *ICH Q6B, Specifications: test procedures and acceptance criteria for biotechnological/biological products*, no. 1, pp. 1–13, 2001, [Online]. Available: http://www.pmda.go.jp/ich/q/q6b_01_5_1e.pdf.

[36] H. Zhu and R. Wang, “A Survey to Design Privacy Preserving Protocol Using Chaos Cryptography,” *International Journal of Network Security*, vol. 20, no. 2, pp. 313–322, 2018, doi: 10.6633/IJNS.201803.20(2).12.

[37] D. Dharminder and P. Gupta, “Security analysis and application of Chebyshev Chaotic map in the authentication protocols,” *International Journal of Computers and Applications*, vol. 43, no. 10, pp. 1095–1103, 2021, doi: 10.1080/1206212X.2019.1682238.

[38] T. T. K. Hue, T. M. Hoang, and A. Braeken, “Lightweight signcryption scheme based on discrete Chebyshev maps,” *2017 12th International Conference for Internet Technology and Secured Transactions, ICITST 2017*, pp. 43–47, 2018, doi: 10.23919/ICITST.2017.8356343.

[39] S. Chatterjee, S. Roy, A. K. Das, S. Chattopadhyay, N. Kumar, and A. V. Vasilakos, “Secure Biometric-Based Authentication Scheme Using Chebyshev Chaotic Map for Multi-Server Environment,” *IEEE Transactions on Dependable and Secure Computing*, vol. 15, no. 5, pp. 824–839, 2018, doi: 10.1109/TDSC.2016.2616876.

[40] H. Zhu, Y. Zhang, and Y. Sun, “Provably secure multi-server privacy-protection system based on chebyshev chaotic maps without using symmetric cryptography,” *International Journal of Network Security*, vol. 18, no. 5, pp. 803–815, 2016, doi: 10.6633/IJNS.201609.18(5).01.

[41] T. W. Lin and C. L. Hsu, “Anonymous group key agreement protocol for multi-server and mobile environments based on Chebyshev chaotic maps,” *Journal of Supercomputing*, vol. 74, no. 9, pp. 4521–4541, 2018, doi: 10.1007/s11227-018-2251-7.

[42] C. T. Li, T. Y. Wu, and C. M. Chen, “A provably secure group key agreement scheme with privacy preservation for online social networks using extended chaotic maps,” *IEEE Access*, vol. 6, pp. 66742–66753, 2018, doi: 10.1109/ACCESS.2018.2879271.

[43] M. Bayat, M. B. Atashgah, and M. R. Aref, “A Secure and Efficient Chaotic Maps Based Authenticated Key-Exchange Protocol for Smart Grid,” *Wireless Personal Communications*, vol. 97, no. 2, pp. 2551–2579, 2017, doi: 10.1007/s11277-017-4623-3.

[44] P. Chen, X. Liu, J. Zhang, C. Yu, H. Pu, and Y. Yao, “Improvement of PRIME Protocol Based on Chaotic Cryptography,” *2019 22nd International Conference on Electrical Machines and Systems, ICEMS 2019*, pp. 1–5, 2019, doi: 10.1109/ICEMS.2019.8922068.

[45] J. Cui, Y. Wang, J. Zhang, Y. Xu, and H. Zhong, “Full Session Key Agreement Scheme Based on Chaotic Map in Vehicular Ad Hoc Networks,” *IEEE Transactions on Vehicular Technology*, vol. 69, no. 8, pp. 8914–8924, 2020, doi: 10.1109/TVT.2020.2997694.

[46] J. Li, L. Wang, L. Wang, X. Wang, Z. Huang, and J. Li, “Verifiable Chebyshev maps-based chaotic encryption schemes with outsourcing computations in the cloud/fog scenarios,” *Concurrency Computation* , vol. 31, no. 22, pp. 1–10, 2019, doi: 10.1002/cpe.4523.

[47] C. Meshram, C. C. Lee, A. S. Ranadive, C. T. Li, S. G. Meshram, and J. V. Tembhurne, “A subtree-based transformation model for cryptosystem using chaotic maps under cloud computing environment for fuzzy user data sharing,” *International Journal of Communication Systems*, vol. 33, no. 7, pp. 1–15, 2020, doi: 10.1002/dac.4307.

[48] N. Tahat, E. S. Ismail, and A. K. Alomari, “Partially blind signature scheme based on chaotic maps and

factoring problems," *Italian Journal of Pure and Applied Mathematics*, no. 39, pp. 165–177, 2018.

[49] L. Zhang, "Cryptanalysis of the public key encryption based on multiple chaotic systems," *Chaos, Solitons and Fractals*, vol. 37, no. 3, pp. 669–674, 2008, doi: 10.1016/j.chaos.2006.09.047.

[50] K. Chain and W. C. Kuo, "A new digital signature scheme based on chaotic maps," *Nonlinear Dynamics*, vol. 74, no. 4, pp. 1003–1012, 2013, doi: 10.1007/s11071-013-1018-1.

[51] J. Kelsey, B. Schneier, D. Wagner, and C. Hall, "Cryptanalytic Attacks on Pseudorandom Number Generators," *Proceedings of the 5th International Workshop on Fast Software Encryption*, pp. 168–188, 1998, doi: 10.1007/3-540-69710-1_12.

[52] M. Tao, K. Ota, M. Dong, and Z. Qian, "AccessAuth: Capacity-aware security access authentication in federated-IoT-enabled V2G networks," *Journal of Parallel and Distributed Computing*, vol. 118, pp. 107–117, 2018, doi: 10.1016/j.jpdc.2017.09.004.

[53] Y. He, F. R. Yu, Z. Wei, and V. Leung, "Trust management for secure cognitive radio vehicular ad hoc networks," *Ad Hoc Networks*, vol. 86, pp. 154–165, 2019, doi: 10.1016/j.adhoc.2018.11.006.

[54] K. Mannix, A. Gorey, D. O'Shea, and T. Newe, "Sensor Network Environments: A Review of the Attacks and Trust Management Models for Securing Them," *Journal of Sensor and Actuator Networks*, vol. 11, no. 3, 2022, doi: 10.3390/jsan11030043.

[55] F. G. Ghajar, J. S. Sratakhti, and A. Sikora, "SBTMS: Scalable blockchain trust management system for VANET," *Applied Sciences (Switzerland)*, vol. 11, no. 24, 2021, doi: 10.3390/app112411947.

[56] Y. Wang, H. T. Luan, Z. Su, N. Zhang, and A. Benslimane, "A Secure and Efficient Wireless Charging Scheme for Electric Vehicles in Vehicular Energy Networks," *IEEE Transactions on Vehicular Technology*, vol. 71, no. 2, pp. 1491–1508, 2022, doi: 10.1109/TVT.2021.3131776.

[57] H. Li, G. Dán, and K. Nahrstedt, "Proactive key dissemination-based fast authentication for in-motion inductive EV charging," in *IEEE International Conference on Communications*, 2015, vol. 2015-Septe, pp. 795–801, doi: 10.1109/ICC.2015.7248419.

[58] M. Pazos-Revilla, A. Alsharif, S. Gunukula, T. N. Guo, M. Mahmoud, and X. Shen, "Secure and Privacy-Preserving Physical-Layer-Assisted Scheme for EV Dynamic Charging System," *IEEE Transactions on Vehicular Technology*, 2018, doi: 10.1109/TVT.2017.2780179.

[59] M. Tajmohammadi, S. M. Mazinani, M. Nikooghadam, and Z. Al-Hamdawee, "LSPP: Lightweight and Secure Payment Protocol for Dynamic Wireless Charging of Electric Vehicles in Vehicular Cloud," *IEEE Access*, vol. 7, pp. 148424–148438, 2019, doi: 10.1109/ACCESS.2019.2946241.

[60] H. Li, G. Dan, and K. Nahrstedt, "Portunes+: Privacy-Preserving Fast Authentication for Dynamic Electric Vehicle Charging," *IEEE Transactions on Smart Grid*, vol. 8, no. 5, pp. 2305–2313, 2017, doi: 10.1109/TSG.2016.2522379.

[61] L. Jiang, S. Xie, S. Maharjan, and Y. Zhang, "Blockchain empowered wireless power transfer for green and secure internet of things," *IEEE Network*, vol. 33, no. 6, pp. 164–171, 2019, doi: 10.1109/MNET.001.1900008.

[62] M. Kim *et al.*, "A secure charging system for electric vehicles based on blockchain," *Sensors (Switzerland)*, vol. 19, no. 13, pp. 1–22, 2019, doi: 10.3390/s19133028.

[63] Z. Xu, F. Li, H. Deng, M. Tan, J. Zhang, and J. Xu, "A blockchain-based authentication and dynamic group key agreement protocol," *Sensors (Switzerland)*, vol. 20, no. 17, pp. 1–19, 2020, doi: 10.3390/s20174835.

[64] H. Tan and I. Chung, "Secure Authentication and Key Management with Blockchain in VANETs," *IEEE Access*, vol. 8, pp. 2482–2498, 2020, doi: 10.1109/ACCESS.2019.2962387.

[65] L. F. A. Roman and P. R. L. Gondim, "Ad Hoc Networks Authentication protocol in CTNs for a CWD-WPT charging system in a cloud environment," *Ad Hoc Networks*, vol. 97, p. 102004, 2020, doi: 10.1016/j.adhoc.2019.102004.

[66] K. Hamouid and K. Adi, “Privacy-aware authentication scheme for electric vehicle in-motion wireless charging,” *2020 International Symposium on Networks, Computers and Communications, ISNCC 2020*, 2020, doi: 10.1109/ISNCC49221.2020.9297199.

[67] W. Ahmed, W. Di, and D. Mukathe, “A Blockchain-Enabled Incentive Trust Management with Threshold Ring Signature Scheme for Traffic Event Validation in VANETs,” *Sensors*, vol. 22, no. 17, 2022, doi: 10.3390/s22176715.

[68] X. Wu, G. Li, and J. Zhou, “A Lightweight Secure Management Scheme for Energy Harvesting Dynamic Wireless Charging System,” *IEEE Access*, vol. 8, pp. 224729–224740, 2020, doi: 10.1109/ACCESS.2020.3044293.

[69] T. Bianchi, S. Asokraj, A. Brighente, M. Conti, and R. Poovendran, “Vulnerability Analysis and Performance Enhancement of Authentication Protocol in Dynamic Wireless Power Transfer Systems,” pp. 1–16, 2022, [Online]. Available: http://arxiv.org/abs/2205.10292.

[70] R. Khalid, O. Samuel, N. Javaid, A. Aldegheishem, M. Shafiq, and N. Alrajeh, “A Secure Trust Method for Multi-Agent System in Smart Grids Using Blockchain,” *IEEE Access*, vol. 9, pp. 59848–59859, 2021, doi: 10.1109/ACCESS.2021.3071431.

[71] K. N. Qureshi, M. N. ul Islam, and G. Jeon, “A trust evaluation model for secure data aggregation in smart grids infrastructures for smart cities,” *Journal of Ambient Intelligence and Smart Environments*, vol. 13, no. 3, pp. 235–252, 2021, doi: 10.3233/ais-210602.

[72] K. Boakye-Boateng, A. A. Ghorbani, and A. H. Lashkari, “A Trust-Influenced Smart Grid: A Survey and a Proposal,” *Journal of Sensor and Actuator Networks*, vol. 11, no. 3, 2022, doi: 10.3390/jsan11030034.

[73] M. Mao, P. Yi, Z. Zhang, L. Wang, and J. Pei, “Roadside Unit Deployment Mechanism Based on Node Popularity,” *Mobile Information Systems*, vol. 2021, 2021, doi: 10.1155/2021/9980093.

[74] D. Dolev and A. C. Yao, “On the Security of Public Key Protocols,” *IEEE Transactions on Information Theory*, vol. 29, no. 2, pp. 198–208, 1983, doi: 10.1109/TIT.1983.1056650.

[75] Z. Xu, X. Li, X. Zhao, M. H. Zhang, and Z. Wang, “DSRC versus 4G-LTE for Connected Vehicle Applications: A Study on Field Experiments of Vehicular Communication Performance,” *Journal of Advanced Transportation*, vol. 2017, pp. 1–10, 2017, doi: 10.1155/2017/2750452.

[76] D. He, S. Chan, and M. Guizani, “Handover authentication for mobile networks: Security and efficiency aspects,” *IEEE Network*, vol. 29, no. 3, pp. 96–103, 2015, doi: 10.1109/MNET.2015.7113232.

[77] A. Kumar and H. Om, “Handover Authentication Scheme for Device-to-Device Outband Communication in 5G-WLAN Next Generation Heterogeneous Networks,” *Arabian Journal for Science and Engineering*, vol. 43, no. 12, pp. 7961–7977, 2018, doi: 10.1007/s13369-018-3255-6.

[78] M. H. Han, S. Liu, S. Ma, and A. Wan, “Anonymous-authentication scheme based on fog computing for VANET,” *PLoS ONE*, pp. 1–20, 2020, doi: https://doi.org/10.1371/journal.pone.0228319.

[79] B. L. Parne, S. Gupta, and N. S. Chaudhari, “SEGB: Security Enhanced Group Based AKA Protocol for M2M Communication in an IoT Enabled LTE/LTE-A Network,” *IEEE Access*, vol. 6, pp. 3668–3684, 2018, doi: 10.1109/ACCESS.2017.2788919.

[80] T. Gunasekhar, K. T. Rao, and M. T. Basu, “Understanding insider attack problem and scope in cloud,” *IEEE International Conference on Circuit, Power and Computing Technologies, ICCPCT 2015*, no. March 2015, 2015, doi: 10.1109/ICCPCT.2015.7159380.

[81] “The AVISPA Project: European Union in the Future and Emerging Technologies (FET Open)-AVISPA v1. 1 user manual,” *2006*. http://www.avispa-project.org.

[82] L. F. A. Roman and P. de Lira Gondim, “Authentication protocol built from a chaotic cryptosystem for a fog and cloud-based CWD-WPT charging station,” *2021 International Wireless Communications and*

Mobile Computing, IWCMC 2021, pp. 370–375, 2021, doi: 10.1109/IWCMC51323.2021.9498756.

[83] H. H. Kilinc and T. Yanik, "A survey of SIP authentication and key agreement schemes," *IEEE Communications Surveys and Tutorials*, vol. 16, no. 2, pp. 1005–1023, 2014, doi: 10.1109/SURV.2013.091513.00050.

[84] J. L. Li, W. G. Zhang, S. Kumari, K. K. R. Choo, and D. Hogrefe, "Security analysis and improvement of a mutual authentication and key agreement solution for wireless sensor networks using chaotic maps," *Transactions on Emerging Telecommunications Technologies*, vol. 29, no. 6, pp. 1–17, 2018, doi: 10.1002/ett.3295.

[85] A. S. Anakath, S. Rajakumar, and S. Ambika, "Privacy preserving multi factor authentication using trust management," *Cluster Computing*, vol. 22, no. s5, pp. 10817–10823, 2019, doi: 10.1007/s10586-017-1181-0.

[86] R. Sugumar, A. Rengarajan, and C. Jayakumar, "Trust based authentication technique for cluster based vehicular ad hoc networks (VANET)," *Wireless Networks*, vol. 24, no. 2, pp. 373–382, 2018, doi: 10.1007/s11276-016-1336-6.

APPENDIX A PUBLICATION IN THE AD HOC NETWORKS JOURNAL.

Ad Hoc Networks 97 (2020) 102004

Contents lists available at ScienceDirect

Ad Hoc Networks

journal homepage: www.elsevier.com/locate/adhoc

Authentication protocol in CTNs for a CWD-WPT charging system in a cloud environment

Luis F.A. Roman*, Paulo R.L. Gondim

Departamento de Engenharia Elétrica, Universidade de Brasília (UnB), Brasília, Brazil

ARTICLE INFO

Article history:
Received 8 April 2019
Revised 3 July 2019
Accepted 2 September 2019
Available online 5 September 2019

Keywords:
Authentication
CTNs
EVs
Fog Computing
VANET
CWD

ABSTRACT

The Internet of Things (IoT) has developed very rapidly in recent years, becoming increasingly complex. The communication things networks (CTNs) is a very important element in IoT for the interconnection of several objects (between objects or objects with the internet). In the context of electric vehicles, the development of CTNs represents a pillar for the implementation of new services such as charge while driving (CWD) based on wireless power transfer (WPT) technology. Cloud-based vehicular ad-hoc networks (VANETs) are one of the networks that can support the high mobility, low latency and connectivity required for a CWD-WPT system. The CWD-WPT charging system provides comfort and time optimization for users if the privacy, integrity and availability of the system are guaranteed. This paper proposes an authentication protocol that uses different cryptographic schemes for key management and distribution in a CWD-WPT cloud charging system that guarantees message privacy and integrity, mutual authentication of system elements and anonymity of EVs.

1. Introduction

Research on the Internet of Things (IoT) has been substantially increased in recent years because of the applications and capabilities it offers. IoT is a complex and heterogeneous ecosystem that interconnects several objects on a large scale to offer innovative services, such as drone-based services, health care services, smart grid features and electric vehicles [1,2].

The widespread adoption of IoT depends on communicating things networks (CTNs), which must adapt to new quality of service (QoS) requirements, carry large volumes of data, and support heterogeneity in different traffic pattern devices for ensuring a reliable delivery of services.

The benefits of IoT and CTN technologies can reach several application areas. Among these areas, we observe that the popularity of electric vehicles (EVs) has grown over the past years, mainly due to the scarcity of fossil fuels and for environmental reasons. According to the Organization for Economic Cooperation and Development, the transportation sector consumes over 50% of the world's oil and is responsible for the emission of approximately 20% of carbon dioxide worldwide. Although the adoption of EVs can improve the environment and reduce the oil dependency, several technological and operational challenges must be overcome [3,4].

One of such challenges is the long duration of charging and battery life, which generate time and mobility restrictions for users [5]. Researchers have been working on the development of a new method of charge while driving (CWD) based on wireless power transfer (WPT) technology. Installed in strategic places, it ensures the EV can travel further and in less time (by popping the charging time) [4,6–8].

The dynamic charging infrastructure or CWD-WPT consists of a series of charging coils called pads embedded in the road pavement. Unlike long and continuous static charging, CWD-WPT comprises a large number of pads that power the EV (micro charging) in only a few milliseconds, depending on the speed of the vehicle [5,6,8].

One of the most outstanding features of the CWD-WPT system is mobility, which promotes changes in the context (location, type of vehicle), access and network connectivity (connection time, wireless or wired network), energy availability (state of charge (SoC)), security and privacy [9].

For the treatment of mobility and connectivity among vehicles, a vehicular ad-hoc network (VANET) is one of the networks that can be considered to support a CWD-WPT system. VANETs have drawn the attention of researchers due to their large variety of applications and services and a safe, efficient, trouble-free and entertaining intelligent transportation system (ITS). VANETs provide vehicles with an onboard communication unit called On-Board Unit (OBU), through which they communicate with both other vehicles and the infrastructure via Roadside Units (RSUs). IEEE 802.11p

* Corresponding author.
E-mail address: lfroman@aluno.unb.br (L.F.A. Roman).

https://doi.org/10.1016/j.adhoc.2019.102004
1570-8705/

standard provides the Wireless Access in Vehicle System (WAVE) protocol and the basic radio standard for dedicated short-range communications (DSRC) at a 5.9 GHz frequency [5,10].

Due to the technological evolution and exponential growth in the number of intelligent vehicles, traditional VANETs have faced flexibility and scalability problems, amongst others. The integration of the cloud with VANET networks seeks to solve the problems of flexibility and scalability, as well as to foster the evolution and creation of new services. Cloud-based VANET communications are comprised of a number of elements and environments that integrate seamlessly to provide users with efficient, scalable and secure services. To achieve this harmonic integration in cloud-based VANET networks, several authors have proposed layered systems with different focuses, where security is a layer that interacts throughout the system [11–13].

Cloud computing is a new paradigm that proposes allocating servers geographically but next the devices to collect, process, organize and store data in real time. Its use in vehicular networks tends to facilitate or provide a great variety of services, besides being a solution to reduce the costs of communication [13]. Cloud computing presents several security challenges, which include data storage, computing, virtualization and network security issues, as well as access control, software security and trust management issues [14].

More specifically, cloud-based vehicular networks security is a challenging problem because of its additional characteristics of heterogeneity and the high volume of vehicles. According to Ziquia et al. [11] the most important security requirements for these networks are: authentication, data integrity, confidentiality, access control, non-repudiation and availability.

The next-generation VANETs must also support high mobility, low latency, real-time services and connectivity, which cannot be provided by conventional cloud computing. An effective solution to vehicular network problems is the fusion of fog computing with cloud computing [10,15], allowing to extend to the edge of wireless networks the conventional paradigm of cloud computing and meeting requirements related to low latency, seamless mobility, data storage close to users and adequate localization of mobile devices. Moreover, the use of fog servers allows better mobility management of vehicles and redirectioning of mobile applications to the closest fog server [15].

Such a cloud environment creates a scalable and hierarchical architecture, which is convenient for the sake of distribute processing and storage capabilities. In our architecture, the company charging server (CCS) is installed in the cloud computing and connected to a group of secondary servers (fog servers - FS), where the fog computing is installed. Each FS groups several RSUs, and each RSU groups several pads together.

The CWD-WPT charging technology in a cloud and fog computing environment can provide comfort and time optimization for EV users, if security, privacy, authentication and anonymity are considered. Mechanisms for EVs to enter a carrier charging service in a controlled and anonymous manner require efficient mutual authentication [16,17].

Proposals for authentication protocols have been presented in the literature. For example the protocols presented by Li et al. [18] and Hussain et al. [19] which focused on the mutual authentication between entity and the preservation of privacy; however, the analysis of security problems is poorly detailed. Other proposals such as those presented by Gunukula et al. [20] and Rabieh and Wei [21] guarantee anonymous authentication, privacy, unlinkability and prevent double spending; however, disregard some attacks that may affect the system. Other shortcomings that the protocols proposed so far have in common is the lack of a formal verification and a comparison of performance with other protocols.

This article proposes a protocol for the administration and distribution of keys in a CWD-WPT charging system in a cloud and fog computing environment, which guarantees privacy and integrity of messages, mutual authentication between the EV and the CWD-WPT charging station and EV anonymity. Its contributions include:

- an authentication and authorization protocol, enabling privacy and integrity preservation as well as key agreement and distribution;
- design of a new CWD-WPT dynamic charging architecture based on a fusion of fog computing with cloud computing;
- preservation of the anonymity of EVs, since the protocol is based on download tickets purchased offline and signed blindly by the system;
- use of cryptographic primitives, such as short signatures and blind signatures based on bilinear pairing for authentication with no jeopardy to the true identity of the EV;
- mutual authentication among the EV and all entities of the CWD-WPT charging station;
- a formal security verification of the protocol by AVISPA tool;
- a security analysis considering several attacks that can affect the system, where a larger number of attacks has been considered, when compared to other proposals (as [18–21]);
- a comparison of performance with other protocols, involving communication and computational costs.

The remainder of the paper is organized as follows: Section 2 addresses related works; Section 3 describes the system model and adversary models; Section 4 provides preliminary information for the understanding of the protocol; Section 5 introduces the protocol; Section 6 reports on performance evaluations and a safety performance analysis; finally, Section 7 is devoted to the conclusions.

2. Related work

Li et al. [18] presented an authentication protocol called "Fast Authentication for Dynamic EV Charging (FADEC)", which has a dedicated short-range communication (DSRC) based on the IEEE 802.11p standard and a five-element architecture, i.e., the utility in charge of the management and administration of the CWD system, a Certification Authority (CA) that certifies all system keys, a set of pads installed on the highway for inducing energy to EVs, RSUs, which are wireless communication devices distributed over the sidewalk and interconnected through a backbone network, and EVs equipped with On-board Units (OBU) that use DSRC to communicate with RSUs.

The authentication protocol was based on the hash-based message authentication code (HMAC), which authenticates entities that rely on a symmetric key shared between two parties, the Elliptic Curve Digital Signature Algorithm (ECDSA), which authenticates vehicle safety messages, and Just Fast Keying (JFK), a key exchange protocol based on the Diffie-Hellman protocol. Li et al. [18] do not emphasize the authentication process and establishment of the session key (JFK protocol). The security based on the JFK protocol has some flaws, since it does not protect the privacy of the user and is susceptible to repetition attacks.

Hussain et al. [19] designed a mutual authentication protocol that ensures privacy for a CWD system via charging plates (CPLs) installed under boards. The authors adopted the concept of online electric vehicle (OLEV) used in South Korea to name vehicles that receive an electric charge from the power line installed below the road surface. The network model is based on a typical VANET network consisting of EVs equipped with an OBU to communicate with the charging infrastructure via DSRC and a tamper-resistant module (TRM) that stores the confidential information of the EV; CPLs installed on the surface of the road and responsible for the

EV charging, VANET Authority, responsible for the registration and revocation of the system, and charging service providing authority (CSPA), responsible for delivering power to the CPs. The Department of Motor Vehicles (DMV) is at the top of the hierarchy, where each VANET Authority must be registered.

The protocol of Hussain et al. [19] uses the following cryptographic primitives to ensure protocol security: El Gamal encryption algorithm over elliptic curve cryptography (ECC), hash, hash chain, and XOR functions, for security analysis, which prove the resistance of the protocol against replaying attacks and impersonation, and dispute resolutions between EVs and the charging system. They have focused only on efforts to ensure mutual authentication and have not analyzed other security issues that may affect the system, such as integrity, DoS attack, Man-In-the-Middle attack, amongst others.

Gunukula et al. [20] designed a protocol that preserves the security of the dynamic charging system and payment of the service. The network model considered in Gunukula et al. [20] is composed of a bank responsible for the sales of charging coins and verification of the validity of currencies. A carrier service provider (CSP) manages the RSU group that is part of the charging station, the RSUs responsible for the management of the group of charging pads installed on the highway, and the charging pads responsible for the induction of energy to the EV.

Towards guaranteeing the security of the system, the protocol was based on the following cryptographic primitives: ECC-based partial blind signature, Diffie-Hellman-based key agreement in ECC, Exclusive-OR and modified hash chain. The safety analysis describes the protocol of Gunukula et al. [20], which guarantees the anonymous authentication of the EV prior to the charging and disassociation of the EV with the currencies purchased. It also provides a description of resistance to attacks such as double spending, man-in-the-middle, and others related to payment for the service; however, it does not analyze attacks that can affect the overall system.

Rabieh and Wei [21] proposed an efficient authentication protocol that guarantees the privacy of drivers. It is composed of EVs that use the charging system, and a harging management center (CMC), i.e., the main component of the architecture, controls the charging controllers and the charging pad (CP). The CPs are installed under the road and induce electric charge to the EVs. A charging controller is installed next to the highway and interconnects the CMC and the pads of the charging station. Finally, the charging carrier implements the necessary infrastructure for charging the EVs (CMC, charging controllers and CPs).

The protocol guarantees the security of client information through the following cryptographic primitives: hash chain, hash, Exclusive-OR operations and blind signatures based on bilinear pairing. The security analysis describes the way the protocol performs a mutual authentication between the EVs and the system and guarantees the privacy of the EVs, unlinkability, double spending and anonymity of the EVs. Differently from other protocols, the one designed by Rabieh and Wei [21] considers an specific architecture of VANET and the security analysis does not consider several attacks that can affect the system such as injection, known key and impersonation attacks, among others.

Laporte et al. [22] described an experimental investigation for characterizing the actual performance of a WPT charging system for EV, in order to carry out a feasibility analysis of the wireless charging technologies that extend the distance traveled by the EV. The work also describes multidisciplinary technical challenges that must be solved, for example, controlling the speed of the EV in the road of load, energy efficiency of transmission from the pads to the EV, and the impact that the WPT system has in the power network.

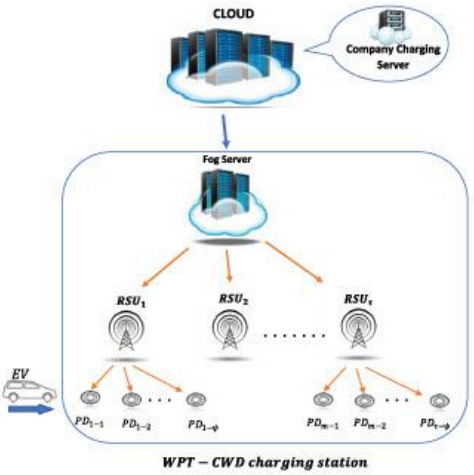


Fig. 1. Network model.

In the paper by Roberts et al. [23], the authors analyzed the high economic costs of the load infrastructure and the potential problems in the power system caused by the high levels of penetration of the EVs. To solve these problems, they proposed a ubiquitous charging system based on a vehicle-to-vehicle (V2V) energy transfer and an certificateless authentication protocol between supplier and customer for a system of Vehicle to Vehicle loads (V2V).

3. Network model and adversary model

This section describes the network and adversary models considered in our study.

3.1. Network model

Fig. 1 shows the network model with company charging service (CCS) (located in a cloud), EVs and a WPT-CWD charging station. Each WPT-CWD charging station is comprised of a fog server, multiple RSUs and charging pads.

The system is assumed to have several WPT-CWD charging stations that communicate with the CCS. EVs can communicate with the CCS via the Internet. RSUs are access points installed on the roadside of the WPT-CWD charging station and can cover several kilometers. We consider there are "τ" RSUs for one WPT-CWD charging station, and each RSU can communicate with a group of "ψ" pads, while the fog server can communicate with all RSUs of the WPT-CWD charging station. Pads are elements that induce an electric charge to the EVs in motion using WPT. Each pad is activated through the validation of a unique key delivered by an EV. EVs can communicate with FS and RSUs through wireless networks, and with the pads through a short-range wireless communication device. Table 1 shows a comparison of the entities of different architectures that support the WPT-CWD service.

3.2. Adversary (attack) model

The Dolev–Yao attack (adversary) model [24] is adopted; in this sense, inspite of messages that can be composed and replayed by

Table 1
Comparison among entities and primitives.

	Entities considered		Cryptographic primitives	Cloud based?	Formal verification security?	Comparison with other protocols?
Li et al. [18]	EV, Pad, RSU, Utility, CA	5 entities	HMAC, symmetric key; ECDSA and Just Fast Keying (JFK)	Not	Not	Not
Hussain et al. [19]	EV, CP, CSPA, VANET Authority	4 entities	ElGamal over ECC, hash, hash chain, and XOR functions,	Not	Not	Not
Gunukula et al. [20]	EV, CSP, RSU, Pad, Bank	5 entities	ECC-based partial blind signature, Diffie-Hellman key agreement based on ECC, XOR and modified hash chain	Not	Not	Not
Rabie et al. [21]	EV, Pad, C-Company, CMC, C-controller,	5 Entities	hash chain, hash, Exclusive-OR operations and blind signatures based on bilinear pairing	Not	Not	Not
Proposed protocol	EV, Pad, RSU, Fog Server, Cloud(CCS)	5 Entities	Diffie-Hellman Key Agreement based on ECC, Short Signatures and Blind signatures, bilinear pairing and Hash Chain,	Yes	Yes	Yes

an adversary, he/she cannot decipher them without knowing the correct cryptographic keys. Moreover, one-way functions are considered unbreakable.

In the proposed scheme, only the CCS entity is trustworthy regarding the real identity of the EV (for collecting tickets). The fog server, the RSU, and the charging pads should do not reveal the real identity of the EV or its owner. Although trustworthy, EVs are curious about private information from the other EVs (SoC, Drivers' identities, etc.), but they do not disturb the operation of the system.

The VANET infrastructure is assumed secure and the RSU has a private key X_{RSU} and a public key Y_{RSU}. The RSUs are connected to the fog server and have a group key K_{G-RSU}. On the other hand, the pads are connected to the RSUs. Finally, a group key for the pads K_{G-pads} is defined.

4. Preliminaries

4.1. Bilinear pairing

Bilinear pairing is defined as the projection of two points of additive set G_1 formed by points on an elliptic curve E of order $l \in Z_p^+$, towards a same point of a multiplicative set G_2 formed by the elements of order $l \in Z_p^+$. The discrete logarithm problem (DLP) is assumed hard in both G_1 and G_2. A mapping $\hat{e} = (G_1, +)^2 \rightarrow (G_2, \cdot)$ satisfies the following properties for all a, b$\in Z_q^*$ and $c, d \in G$ ([25]).

(1) Bilinear:

$$\hat{e}(a+c,d) = \hat{e}(c,d)\hat{e}(a,d)$$

$$\hat{e}(c,d+a) = \hat{e}(c,d)\hat{e}(c,a)$$

(2) Non-degenerative:

$$\hat{e}(c,d) \neq 1_{G_2}$$

(3) Computationally efficient.

Bilinear pairings have other easily verifiable properties, such as:

(1) $\hat{e}(x, \infty) = 1$ e $\hat{e}(\infty, x) = 1$
(2) $\hat{e}(c,-d) = \hat{e}(-d,c) = \hat{e}(d,c)^{-1}$
(3) $\hat{e}(ac, bd) = \hat{e}(d,c)^{ab}$
(4) $\hat{e}(c,d) = \hat{e}(d,c)$,

and can be used for data encryption, digital signatures and key agreements. In our protocol they are employed for the generation of digital signatures.

4.2. Digital signatures

A digital signature is one of the most important cryptography-based resources. It indicates the owner or creator of a document or clarifies someone agrees on the content of a document. Some digital signatures are based on a public key that links the identity of the user with its public key, whereas others are based on the identity of the that generates the public key from the user's identity through a deterministic algorithm. The public key verification is based on the use of the user's identity, making this scheme more efficient. The first short bilinear pairing scheme was created by Boneh et al. [26], and from it were created a large number of signature schemes based on the coincidence for different applications [25]. Below is a description of the digital signature schemes used in our protocol.

4.2.1. Short signatures

Short signatures work well in environments of memory and bandwidth restrictions. The most used signature schemes are RSA (Rivest, Shamir and Adleman) and DSA (Digital Signature Algorithm), however, the signatures they generate are long. For example, if the 1024-bit module is used, the signatures of RSA and DSA are 1024 bits long. The bilinear pairing scheme provides short-length signatures of approximately 160 bits with a security level similar to those of 1024-bit RSA and DSA signatures [25].

A signature scheme based on bilinear pairing commonly involves [25]:

- Initialization: Let H: $\{0, 1\}^* \rightarrow G1$ be a map-to-point hash function. The secret key is $X, \in Z_q^*$, and the public key is $Y = X^*P$ for a signer.
- Sign: Given secret key x and a message $m \in \{0, 1\}^*$, compute signature $\sigma = X^*H(m)$
- Verify: Given public key $Y = X^*P$, a message m and a signature σ, verify $e(P, \sigma) = e(Y, H(m))$.

4.2.2. Blind signatures

Blind signatures have been widely used in digital payment schemes for the obtaining of the signature of a document without

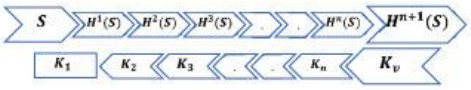


Fig. 2. Hash chain.

the signatory knowing the information of the document. Moreover, the user cannot obtain other valid signatures of the same document after an interaction with the subscriber. The scheme used for our protocol was created by Zhang et al. [27] and is called "ID-Based Blind Signature and Ring Signature from Pairings". It is characterized by the use of an identity-based cryptosystem over bilinear pairings for the verification and authentication of the signed information without knowing the identity of the sender.

4.3. Hash chain

Hash chain is a computational operation for the efficient authentication of one-time passwords, extending the lifetime of digital certificates, building one-time signatures, amongst other functions. It was used in this study for the authentication and creation of session keys [28].

A hash chain is generated by a hash algorithm, as SHA (Secure Hash Algorithm), through which a user randomly selects a seed (S) and calculates the entire key chain. Fig. 2 shows the process of creation of keys with a chain hash.

The keys generated must be used in the opposite order of their generation, i.e., the last generated key K_n must be the first one used and the first key K_1 must be the last key used, such that an attacker listening to the channel cannot calculate a valid key from a used one. In our protocol, a public verification key K_v, is calculated applying $n+1$ hashes to S for the validation of the keys. To verify a hash chain, an entity only applies successive hashes until it reaches the value of key K_v. If the key received after the application of n hash at maximum is not given the same value of the verification key, it is discarded.

5. Proposed protocol

Our protocol is divided into four phases, namely initialization, registration, ticket purchasing and charging request (see Fig. 3). In the initialization phase, sets, functions and master keys necessary for the start of the operation of the scheme are defined. In the registration phase, the data of the EV are stored in the system. In the phase of purchasing tickets, EVs purchase one or several tickets to perform the EV charge in the charging station. Finally, in the charging request phase, the delivery, validation, authentication and generation of keys necessary for the charging of EV through the WPT-CWD system are performed.

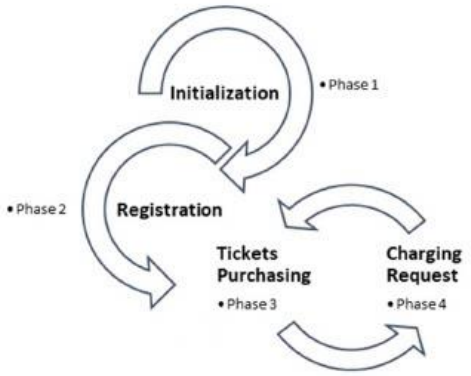


Fig. 3. Phases of the proposed protocol.

1st phase: **Initialization of the System**

In this phase, the use of the pseudorandom random number generator (PRNG) is considered for the generation of nonces and seeds. The PRNG will be reinitialized at random times, and the random value generated by the PRNG will be processed by a hash function to be used by the system. In PRNG, the initial state is changed with parameters that are the product of applying hash functions over input values concatenated with timestamps [29].

The system had chosen two cyclic groups G_1 and G_2 of orders q and P and a generator element of group G_1 are chosen. G_1 and G_2 are supposedly related to a non-degenerative pairing and a bilinear map that can be efficiently computed:

$\hat{e}$: $G_1 \times G_1 \rightarrow G_2$ such that $\hat{e}(P, P) \neq 1G_2$ and $\hat{e}(aP,bQ) = \hat{e}(b\ P,a\ Q) = \hat{e}(P_1, Q_1)^{ab} \in G_2$ for every a, b $\in Z_q^*$ and every P, Q $\in G_1$. Moreover, the hash functions of the system are defined: H: $\{0, 1\}^* \rightarrow G_1$ and $H_1 : \{0, 1\}^*.G \rightarrow Z_q^*$.

CCS then chooses a master private key Y_{ccs}, $\in Z_q^*$ and calculates its global public key $Y_{pub} = X_{ccs}{}^*P$. Additionally, it computes its own public key $Q_{ccs} = H(ID_{ccs})$ and private key $S_{ccs} = X_{ccs}{}^*Q_{ccs}$.

Finally, the company charging center (CCS) defines an elliptical curve on a finite field E (Fq) and parameters $\{G_1, G_2, \hat{e}, P, H, H_1, P_{pub}, Q_{ccs}\}$ are published.

2nd phase: **EV registration**

All owners of EVs who want to use the CWD charging system register with the CCS through a secure channel. The user chooses a random number X_{EV}, $\in Z_q^*$ and calculates $Y_{EV} = X_{EV}{}^*P$, where X_{EV} will be his/her private key and Y_{EV} will be the public key. This public key along with identity (ID_{EV}) and vehicle charging parameters (VCP) are sent to the CCS to be stored. Finally, the CCS creates a certificate $Cert_{EV} = X_{ccs}{}^*Q_{EV}$ where $Q_{EV} = H(ID_{EV})$ and sends it to the EV.

3rd phase: **Tickets Purchasing**

Each ticket is assumed to have a specified amount of energy to be induced to the EV through a certain number of pads. The tickets are purchased through a secure channel and the EV has an associated bank account in the CCS, with enough money for their purchase..

The first message, m_1, requesting the purchase of n tickets to the CCS is sent by the EV.

$$m_1 = \{n, ID_{EV}, Cert_{EV}\}$$

The CCS receives it and generates n random values $\{r_1, r_2, \ldots, r_n\} \in Z_q^*$. For each r_i for $0 \leq i \leq n$, $R_i = r_i{}^*P$ is calculated and a message m_2 containing set $R = \{R_1, R_2, \ldots, R_n\}$ is sent to the EV:

$$m_2 = \{R\}$$

The EV receives it, creates n random pseudonyms $\{PID_1, PID_2, \ldots, PID_i, \ldots, PID_n\}$, and applies a blind signature to each n PID. It then chooses two random numbers a, $b \in Z_q^*$ and computes the blind pseudonym (B) for every pseudonym PID:

$$B_i = H\left(PID_i, \hat{e}\left(bQ_{ccs} + R_i + aP, Y_{pub}\right)\right) + b$$

The EV sends message m_3 with the $B = \{B_1, B_2, \ldots, B_i, \ldots B_n\}$ to the CCS to receive the system signature.

$$m_3 = \{B\}$$

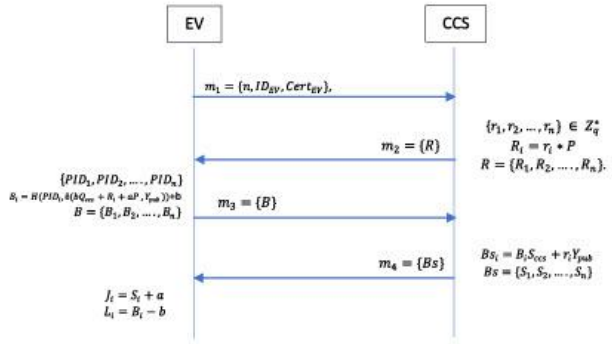


Fig. 4. Ticket purchasing.

The CCS receives the message and signs all blind pseudonyms from set B:

$$Bs_i = (B_i * S_{ccs}) + (r_i * Y_{pub})$$

It then sends message m_4 ($Bs = \{Bs_1, Bs_2, \ldots, Bs_n\}$) to the EV.

Finally, the EV receives m_4 containing set Bs and calculates two values (J and L) for signature verification to obtain the signature of each blind pseudonym set $B = \{B_1, B_2, \ldots, B_i, \ldots B_n\}$:

$J_i = Bs_i + aY_{pub}$, and $L_i = B_i - b$, therefore, the signature of each blind pseudonym B_i will be the pair of values (J_i, L_i). The Fig. 4 shows a summary of the ticket purchase phase and a summary of the ticket purchasing phase, respectively.

4th phase: **Charging Request**

This phase describes the verification, authentication, and creation of session keys between the EV and the WPT-CWD charging station.

Once the EV owner has a valid ticket (PID_1, J_1, L_1) and wants to charge his/her EV in a WPT/CWD charging station, the EV system selects a random number $\varphi_{EV} \in Z_q^*$, calculates $\phi_{EV} = \varphi_{EV} * P$, and sends an m_1 message to the fog server

$$m_1 = \{\phi_{EV}, t_5, H(\phi_{EV} \| t_5)\}, \text{ where } t_5 \text{ is a timestamp.}$$

The fog server checks the hash and message timestamp m_1. If it succeeds, the server chooses a random value $\varphi_{fs} \in Z_q^*$ and calculates session $k_{fs-EV} = \varphi_{fs} * \phi_{EV}$ and values, such that the EV can calculate session key $\phi_{fs} = \varphi_{fs} * P$, verification key $VK = H(k_{fs-EV})$, and signature message $\sigma_{fs} = x_{fs} * H(\phi_{fs}, CK, t_5)$. The fog server immediately sends message m_2 to the EV.

$$m_2 = \{\phi_{fs}, VK, t_6, \sigma_{fs}\}$$

When $m_2' = \{\phi_{fs}', VK', t_6', \sigma_{fs}'\}$ arrives, the EV checks fog server's signature σ_{fs}': $\hat{e}(\sigma_{fs}', P) = ?\hat{e}(H(\phi_{fs}', VK', t_6'), Y_{fs})$. If the equality is successful, the EV authenticates the fog server, uses the message values to calculate session key $k_{fs-EV} = \varphi_{EV} * \phi_{fs}$, and verifies the integrity of the key calculating $VK = H(k_{fs-EV})$ and checking if $VK' = ?VK$. If the equality is successful, the EV uses the session key to crypt and send message m_3 containing the ticket (PID_1, J, L) and a timestamp to the fog server.

$$m_3 = \{PID_1, J_1, L_1, t_7\}_{k_{fs-ev}}$$

When the message arrives at the fog server, it is deciphered with session key k_{fs-EV}, the timestamp is checked and the pseudonym validity is immediately verified: $L_i = H(PID_i, \hat{e}(J_i, P)\hat{e}(Q_{ccs}, Y_{ccs})^{-L_i})$. If the validation is successful, the fog server chooses random seeds α_1, α_2, creates a new pseudonym $PID2_1 = H_1(PID_1 + \alpha_1)$, and sends an encrypted message m_4 containing seed α_1, τ and a timestamp to the EV. A message broadcast m_5 encrypted with key K_{G-RSU} and containing seeds α_1, α_2, τ and a timestamp is also sent to the group of RSUs. Finally, the fog server revokes pseudonym PID_1 to prevent its reuse.

$$m_4 = \{\alpha_1, \tau, PID2_1, t_8\}_{k_{fs-ev}}, \text{ sent to EV}$$

$$m_5 = \{\alpha_1, \alpha_2, \tau, PID2_1, t_9\}_{k_{G-RSU}}, \text{ sent to RSU}$$

When the EV receives m_4, it decrypts it and checks its timestamp. If the verification is successful, it calculates, offline, a verification key for each RSU using a hash chain $H^{RSU}(\alpha_1) = \{H(\alpha_1), H^2(\alpha_1), \ldots H^\tau(\alpha_1)\}$. It also calculates, offline, and with each verification key, a message authentication code $HMAC_{RSU}^d = \{PID2_d \| 1 \| t_8 \| H^d(\alpha_1)\}$, and authenticates each RSU.

All RSUs receive the message m_5 from the fog server, decrypt with the group key (k_{RSU-G}) and check the timestamp. If the check succeeds, each RSU calculates a check key $H^d(\alpha_1)$, a session key $k_{RSU-PID2} = H(H^d(\alpha_1 \| d) \oplus H^d(\alpha_2))$, a verification key ($VK$) and a message authentication code $HMAC_{RSU}^d = H(H^d(\alpha_2) \| VK_2 \| t_{10} \| H^d(\alpha_1))$, where d is the position of the RSU at the charging station d: $1 \leq d \leq \tau$.

The authentication of the first RSU is explained in what follows for simplifying the description of the protocol. The authentication of the EV with the other RSUs and the group of pads managed by it undergo the same authentication process.

When the EV is authenticated with the first RSU, it sends a message m_6 containing message pseudonym $PID2_{EV}$, the sequence number of RSU, a timestamp, and an $HMAC_{RSU}^1 = H(PID2_{EV} \| 1 \| t_9 \| H^1(\alpha_1))$.

$$m_6 = \{PID2_{EV}, 1, t_{10}, HMAC_{RSU}^1\}$$

When the message arrives, the RSU checks if its database contains PID_{EV}. If so, it checks $HMAC_{RSU}^1$ with the values associated with $PID2_{EV}$. If the verification is successful, the RSU computes session key $k_{RSU-EV} = H(H^1(\alpha_1 \| 1) \oplus H^1(\alpha_2))$, and sends message m_7 containing a value $H^1(\alpha_2)$, a key verification code $VK_2 = H(k_{RSU-EV})$, and its signature $HMAC_{EV}^1 = H(H^1(\alpha_2) \| VK_2 \| t_{10} \| H^1(\alpha_1))$ to the EV. It also adds the check key to a revocation list of RSUs to prevent reuse of the key.

$$m_7 = \{H^1(\alpha_2), VK_2, t_{11}, HMAC_{EV}^1\}$$

When $m_7' = \{H^1(\alpha_2)', VK_2', t_{10}', HMAC_{EV}^{1'}\}$ arrives, the EV checks the RSU's $C_{EV}^{1'} = ?HMAC_{EV}^{\tau} = H(H^1(\alpha_2)' \| VK_2' \| t_{10}' \| H^1(\alpha_1)$ If the

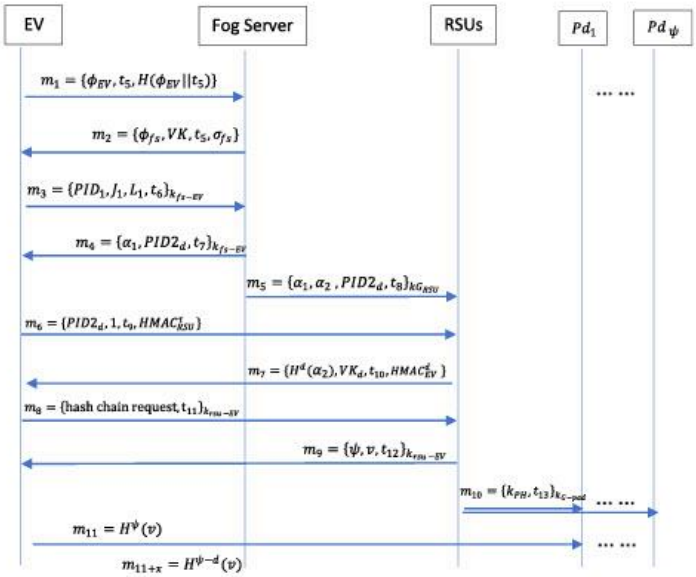


Fig. 5. Charging request phase.

equality is successful, the EV authenticates the RSU and uses the message values to calculate session key $k_{RSU-EV}' = H(H^1(\alpha_1||1) \oplus H^1(\alpha_2)')$. It also verifies the integrity of the key calculating $K_2 = H(k_{rsu-EV}')$, and compares $VK_2' = ?VK_2$. If the equality is successful, the EV uses the session key to send an m_8 message containing a hash chain request to the RSU.

$$m_8 = \{\text{hash chain request}, t_{12}\}_{k_{rsu-ev}}$$

The RSU receives, decrypts, checks the timestamp (t_{12}), and sends message m_9 to the EV. ψ is the number of keys to be authenticated in each pad and $v \varepsilon Z$ is a random number used as the initial value for the calculation of the hash chain. Additionally, the RSU sends all pads a message broadcast m_{10} encrypted with group key (k_{G-pad}) that contains public hash chain verification key $k_{PH} = H^{\psi+1}(v)$ used for the verification of the keys sent by EV.

$$m_9 = \{\psi,\ v,\ t_{13}\}_{k_{rsu-ev}}$$

$$m_{10} = \{k_{PH},\ t_{13}\}_{K_{G-pad}}$$

The EV receives and decrypts m_9 with values ψ and v, and computes hash chain $H^{\psi}(v)$. Each block of pads managed by the RSU receives and decrypts broadcast message m_{10} with the group key. The message contains public hash chain verification key $k_{PH} = H^{\psi+1}(v)$. Whenever a key from a hash chain is sent by the EV (m_{11}) to one of the pads, the pad checks if the key has been validated by iteratively applying $\xi - \psi$ (for $0 \leq \xi \leq \psi + 1$) times the hash function and compares it to the public key hash chain (verification key). If the verification is successful, the pad checks the status of the key in the revocation list. If the key has not been revoked, it accepts the key sent by the EV and revokes it to avoid double use. The process ends when the EV has passed over all pads.

Below is the mathematical proof of the signing blind pseudonym and fog server's signature verification:

- Signing blind pseudonym verification:

$$L = ?H\left(PID, \hat{e}(J,P)\hat{e}(Q_{ccs}, Y_{ccs})^{-L}\right)$$

$$\begin{aligned} L &= H\left(PID, \hat{e}(J,P)\hat{e}(Q_{ccs}, Y_{ccs})^{-L}\right) \\ &= H\left(PID, \hat{e}\left(B.S_{ccs} + r.Y_{pub} + a.Y_{pub}, P\right)\hat{e}(-L.Q_{ccs}, x_{ccs}.P)\right) \\ &= H\left(PID, \hat{e}\left(B.S_{ccs} + r.Y_{pub} + a.Y_{pub}, P\right)\hat{e}(-(B-b).Q_{ccs}, x_{ccs}.P)\right) \\ &= H\left(PID, \hat{e}\left(B.S_{ccs} + r.Y_{pub} + a.Y_{pub}, P\right)\hat{e}((-B+b)(Q_{ccs}.x_{ccs}), P)\right) \\ &= H\left(PID, \hat{e}\left(B.S_{ccs} + r.Y_{pub} + a.Y_{pub}, -B.S_{ccs} + b.S_{ccs}, P\right)\right) \\ &= H\left(PID, \hat{e}\left(r.Y_{pub} + a.Y_{pub} + b.(Q_{ccs} * x_{css}\right), P)\right) \\ &= H\left(PID, \hat{e}\left(b.Q_{ccs} + R_I + a.P, Y_{pub}\right)\right) \end{aligned}$$

- Fog server's signature verification: $\hat{e}(\sigma_{fs}', P) = ?\hat{e}(H(\phi_{fs}', VK', t_5'), Y_{fs})$

$$\begin{aligned} \hat{e}\left(\sigma_{fs}', P\right) &= \hat{e}\left(H\left(\phi_{fs}', VK', t_5'\right), Y_{fs}\right) \\ &= \hat{e}\left(H\left(\phi_{fs}', VK', t_5'\right), x_{fs} * P\right) \\ &= \hat{e}\left(x_{fs} * H\left(\phi_{fs}', VK', t_5'\right), P\right) \\ &= \hat{e}\left(\sigma_{fs}', P\right) \end{aligned}$$

Fig. 5 shows the flow of messages exchanged among the entities in the charging request phase.

6. Security and performance analyses

This section addresses an analysis of the security and performance of the protocol and a comparison with other protocols used for the authentication of a WPT-CWD system.

6.1. Security analysis

6.1.1. Security properties

Below is an analytical description of the security attributes, like mutual authentication, privacy preservation and integrity protection guaranteed by our protocol and a description of the way it resists attacks.

(1) Mutual Authentication: this process is established among FS, RSU and EVs. EVs authenticate FS by verifying message (m_2) signature. FS authenticates the valid ticket of an EV by verifying the blind signature sent in message 3 and using public parameters of the system. The RSU authenticates the EV by calculating the hash of message 6 containing an α_1 (delivered by the FS to the EV in message 4, and the RSU in message 6) sent by the EV. EVs authenticate to RSUs by verifying message 7 HMAC.
(2) Preservation of privacy: the EV identity is kept confidential by the CCS during the purchase of the tickets; FS, RSUs and pads are unable to obtain the user's identity from the ticket. The privacy of the location is also guaranteed, since the tickets and PIDs used by the EV in different locations cannot be correlated with a single vehicle.
(3) Protection to integrity: the integrity of the messages exchanged is maintained with the hash function and digital signatures. The system can identify whether an adversary manipulates the message by verifying the hash function value or the digital signature of the message.
(4) Perfect Forward Secrecy (PFS): the proposed protocol guarantees PFS as follows:
   - In the process of creating session key $k_{(fs-EV)}$ between EV and FS to encrypt the messages, the random elements φ_{EV}, φ_{fs} and a blind message signature are used. Even if the session key $k_{(fs-EV)}$ is compromised, the previous messages cannot be recovered because of the CDH problem;
   - In the process of creating a session key $k_{(RSU-EV)}$ between the EV and the RSU to encrypt the messages, the random elements α_1, α_2 and $PID2_1$ are used. Even if some or all of the random values are committed and the attacker manages to recreate the session key $k_{(RSU-EV)}$, previous messages cannot be recovered due to the CDH problem;
   - In the process of creating the key $H^{\psi(v)}$ between the EV and the Pads, in the worst case when the seed v is compromised, the attacker will not be able to decipher the previous messages;
   - If the CCS (X_{ccs}) private key is compromised, an attacker will not be able to recreate previous session keys and therefore decrypt old messages due to the random values used for generating session keys.

6.1.2. Prevention against attacks

Below are the different types of attacks that can affect the VANET network and a description of the way our protocol can resist them:

- Impersonation: An attacker that aims to enter the system using a false ticket cannot deceive the system, since it cannot sign the ticket correctly. On the other hand, session keys are generated whenever an EV uses a new ticket. It prevents the use of old parameters in other EVs or by itself.
- MITM: The use of digital signatures for the verification of the authenticity and integrity of messages m_2 and m_7 ensures that an MITM attack cannot be succesfull. On the other hand, when the EV performs an authentication process with the RSU, the EV sends a hash chain generated by the seed α_1 in message m_6, taking into account only an authentic EV can generate the valid hash chain, the MITM attack is mitigated.
- Replay and Injection: The use of a timestamp and random numbers in the messages avoids repetitive attacks and hash functions and digital signatures evidence the injection of data in the messages.
- Known key: Our protocol generates tickets which can be used only once. The ticket is added to the revocation list after its validity has been checked. Both system and EV generates random values for to create session keys, i.e., new session keys are generated for every new ticket for EV communication with the charging station, thus preventing an attacker from charging his/her car using old keys they may know.
- DoS: DoS attacks can affect the fog server and RSUs. In the first case, the fog server resists DoS attacks by validating tickets with public system parameters and revocation lists. In the second case, RSUs resist DoS attacks by efficiently validating connection requests using an HMAC code and verifying the auth variable α_1 in the revocation lists. Only users previously authenticated by the fog server have a valid α (**alpha**) to create a valid HMAC. If an attacker attempts to connect to the RSU using an already used HMAC or a false HMAC, the RSU rejects the communication.
- Unlinkability: No entity can link PID_{ev} with a single EV, because the CCS blindly signs this value on the ticket (c_i). Moreover, the fog server checks the PID_{ev} (C', S') signature only with public parameters of the system.
- Double Spending: When an EV uses PID_{ev} and its signatures (C ', S') to authenticate to the fog server, PID_{ev} is revoked and published on a fog server's revocation list. In the authentication process, the fog server checks if PID_{ev} is on the list for terminating the continuing authentication process at the charging station. The same occurs in the EV authentication process in the RSU. PID_{ev} is revoked and published on a revocation list of RSUs.
- Random number leakage attack: to prevent this type of attack, the protocol uses the following operations and controls in relation to the PRNG system [29]:
   - A hash function will be executed on the inputs that are counted with a timestamp;
   - A hash function will be executed on the PRNG outputs;
   - In a period of random time, a new initial PRNG state will be generated;
   - Smart seed will be used at the starting points of the PRNG.
- Privileged insider attack: to prevent this type of attack, the company must establish security policies, internal processes and mechanisms for the prevention and detection of attacks. The following is a set of policies to be implemented in the system to prevent such attacks or mitigate damages [30]:
   - Awareness of security: the company's security policies and procedures must be known to all internal staff and external partners;
   - Classification of duties: it is necessary to classify the duties of employees and employers, to prevent or detect the attacks effectively;
   - Whirling of duties: when you have several important jobs, you should have several employees with the knowledge of the execution of these jobs; in each time period, these officials have to turn to different jobs to avoid malicious actions;
   - Limited privileges: limited access privileges (physical and in systems) must be given to officials to restrict access to confidential information or important company equipment;
   - Encrypt sensitive data: confidential data must be encrypted and stored in secure locations. The company must be backed up in the event that the system data is corrupted;

Table 2
Comparison of security properties.

	Li et al. [18]	Hussain et al. [19]	Gunukula et al. [20]	Rabie et al. [21]	Proposed protocol
Mutual authentication and key agreement	Yes	Yes	Yes	Yes	Yes
Confidentiality	No	Yes	Yes	Yes	Yes
Integrity	Yes	Untreated	Untreated	Untreated	Yes
Privacy	No	Yes	Yes	Yes	Yes
Injection attacks	Untreated	Untreated	Untreated	Untreated	Yes
Forward secrecy	Untreated	Untreated	Untreated	Untreated	Yes
Replay attack	No	Yes	Untreated	Yes	Yes
Known key attack	Untreated	Yes	Untreated	Untreated	Yes
DoS attack	Yes	Untreated	Untreated	Untreated	Yes
Man-in-the-middle attack	Yes	Untreated	Yes	Untreated	Yes
Impersonation attack	Untreated	Yes	Untreated	Untreated	Yes
Unlinkability	Untreated	Untreated	Untreated	Yes	Yes
Double spending	Untreated	Untreated	Yes	Yes	Yes
Random number leakage attack	Untreated	Untreated	Untreated	Untreated	Yes
Privileged insider attack	Untreated	Untreated	Untreated	Untreated	Yes
Masquerade attack	Untreated	Untreated	Untreated	Untreated	Yes

```
role role_EV(EV:agent,FS:agent,RSU:agent,PAD:agent,
             H1:function,H2:function,H3:function,H4:function,
             H5:function,CK:function,Kfsev:symmetric_key,
             Krsuev:symmetric_key,SND,RCV:channel(dy))
played_by EV
def=
      local
       State:nat,T5:text,Sigfs:text,T6:text,Vfifs:text,Vfiev:text,
       C:text,PID:text,S:text,T7:text,Tao:text,T8:text,Y:text,
       PID2:text,T10:text,HMAC:function,Sigrsu:text,T11:text,
       Alf1:text,M:function,Alf2:text,P:text,Req:text,T12:text,
       T13:text,Is:text,Psi:text
      init
       State := 0
      transition
       1. State=0 /\ RCV(start) =|> State':=1 /\ T5':=new()
       /\ P':=new() /\ Vfiev':=new() /\ secret(Vfiev',sec_5,{})
       /\ SND(M(Vfiev'.P').T5'.H1(M(Vfiev'.P').T5'))
       2. State=1 /\
RCV(M(Vfifs'.P).T6'.CK(M(Vfifs'.M(Vfiev.P))).Sigfs')
       =|> State':=2 /\ secret(Vfiev',sec_5,{}) /\
secret(Vfifs',sec_6,{})
       /\ T7':=new() /\ C':=new() /\ S':=new() /\ PID':=new()
       /\ SND({PID'.S'.C'.T7'}_Kfsev)
       4. State=2 /\ RCV({Alf1'.Tao'.PID2'.T8'}_Kfsev) =|>
State':=3
       /\ secret(Alf1',sec_1,{}) /\ T10':=new() /\ Y':=new()
       /\ SND(PID2'.Y'.T10'.HMAC(PID2'.Y'.T10'.Alf1'))
       7.                State=3                /\
RCV(M(H3(Alf2').P).T11'.CK(M(Alf1.M(Alf2'.P))).Sigrsu')
       =|> State':=4 /\ witness(EV,RSU,auth_10,Sigrsu')
       /\ secret(Alf2',sec_2,{}) /\ secret(Alf1',sec_1,{})
       /\ T12':=new() /\ Req':=new() /\
SND({Req'.T12'}_Krsuev)
       9. State=4 /\ RCV({Psi'.Is'.T13'}_Krsuev) =|>
State':=5
       /\ secret(Is',sec_4,{}) /\ secret(Psi',sec_3,{})
       /\ SND(H5(Psi'.Is'))
end role
```

Fig. 6. Role of EV in HLPSL.

- Defense in depth: a layered security policy must be implemented, where each layer has specific tasks for system protection.

- Masquerade attack: the proposed protocol is safe against server masking attacks, because an attacker cannot represent the response messages that are sent by the FS or RSU. The FS and RSU sign the contents of the response messages with their private key, so an attacker cannot recreate the signature of the response messages because they do not have the FS or RSU private key.

Table 2 shows a comparison of the security analysis between our protocol and other schemes for authentication for CWD-WPT load stations.

6.1.3. Formal verification of the proposed protocol

The protocol was formally verified by AVISPA, a commonly used tool for security protocol assessments. The entities and message exchanges were described by the HLPSL (High Level Protocol Specification Language) language [31].

AVISPA has four protocol validation modes called "Backends", including On-the-Fly Model Checker (OFMC) and CL-AtSe (Constraint-Based Attack Locator). The results of the verification of a protocol are "SAFE", if no problem has been detected, and "UNSAFE", if an attack has been successful. AVISPA provides a report only when the result is "UNSAFE". The report addresses the successfully executed attack.

Modeling of the proposed protocol in HLPSL. The HLPSL language enables the construction of a protocol model to be evaluated. Figs. 6–8 show some of the HLPSL codes that modelled our protocol.

Fig. 6 displays the HLPSL code that models the behavior of the EV in our protocol. The structure of the code is the same as those of the codes of other entities (CCS, FS and PAD) and consists of the following parts:

- Statement of the agents, communication channels and constants known by the entity;
- Declaration of variables calculated or received by other entities; and
- Statement of the functions to be used.

States are created immediately after the creation of the aforementioned declarations and describe the operations and messages to be exchanged between entities. At the end of each state, the elements that must be kept confidential and authenticated variables are declared.

Fig. 7 shows the HLPSL language code that describes the establishment of the sessions and the environment of the execution of the protocol. The elements (variants, keys, agents, etc.) likely to be acquired by an attacker are also declared.

Finally, Fig. 8 shows the security objectives to be guaranteed by the protocol according to the definition of elements declared as se-

```
role session1(Krsuev:symmetric_key,HMAC:function,KGfsrsu:symmetric_key,
              CK:function,Kfsev:symmetric_key,EV:agent,FS:agent,RSU:agent,
              PAD:agent,H1:function,H2:function,H3:function,H4:function,
              H5:function,KGrsupad:symmetric_key)
def=
    local
     SND4,RCV4,SND3,RCV3,SND2,RCV2,SND1,RCV1:channel(dy)
    composition
     role_PAD(EV,FS,RSU,PAD,H1,H2,H3,H4,H5,KGrsupad,SND4,RCV4)
     /\ role_RSU(EV,FS,RSU,PAD,H1,H2,H3,H4,KGfsrsu,HMAC,Krsuev,KGrsupad,SND3,RCV3)
     /\ role_FS(EV,FS,RSU,PAD,H1,H2,CK,Kfsev,KGfsrsu,HMAC,SND2,RCV2)
     /\ role_EV(EV,FS,RSU,PAD,H1,H2,H3,H4,H5,CK,Kfsev,Krsuev,SND1,RCV1)
end role
```

Fig. 7. Specification of the session role in HLPSL.

```
Goal
    secrecy_of sec_1
    secrecy_of sec_2
    secrecy_of sec_3
    secrecy_of sec_4
    secrecy_of sec_5
    secrecy_of sec_6
    authentication_on auth_7
    authentication_on auth_8
    authentication_on auth_9
    authentication_on auth_10
    authentication_on auth_11

end goal

environment()
```

Fig. 8. Security objectives and related secrets of our protocol in HLPSL.

crets in the functions of the entity and the values that authenticate the entities.

- secrecy_of sec_1: keep secret α_1;
- secrecy_of sec_2: keep secret α_2
- secrecy_of sec_3: keep secret ψ
- secrecy_of sec_4: keep secret υ
- secrecy_of sec_5: keep secret ϕ_{EV}
- secrecy_of sec_6: keep secret ϕ_{fs}
- authentication_on auth_7: EV authenticates FS on σ_{fs};
- authentication_on auth_8: FS authenticates EV on PID_1;
- authentication_on auth_9: RSU authenticates EV on α_1;
- authentication_on auth_10: EV authenticates RSU on $H(\alpha_2)$;
- authentication_on auth_11: PAD authenticates EV on υ;

Security check results. Simulations in AVISPA with OFMC and CL-AtSe backends checked the security of the protocol, which was considered safe for both backends, according to the results (see Fig. 9).

6.2. Performance analysis

This subsection reports on a performance analysis of computational and communications costs. The authentication procedures between the fog server and EVs (FS-EVs), EV and RSUs (EVs-RSU), and EVs and pads (EVs-pads) were assumed independent, since those processes can be conducted in different time periods and locations. For example, an EV can authenticate the fog server far from the charging station with considerable time in advance. The following EVs-RSUs authentication process can be performed hundreds of meters from the first pad and several seconds in advance. Lastly, an EV must be authenticated by the pad a few centimeters from it and microseconds in advance.

6.2.1. Communication costs

We consider that this transmission uses high-coverage communication technology such as LTE, so that the EV is able to perform the exchange of information with the FS before entering the CWD-WPT charging station. For communications within the CWD-WPT

```
% OFMC
% Version of 2006/02/13
SUMMARY
  SAFE
DETAILS
  BOUNDED_NUMBER_OF_SESSIONS
PROTOCOL
  /home/span/span/testsuite/results/hlpslGenFile.if
GOAL
  as_specified
BACKEND
  OFMC
COMMENTS
STATISTICS
  parseTime: 0.00s
  searchTime: 0.24s
  visitedNodes: 11 nodes
  depth: 6 plies
```

```
SUMMARY
  SAFE

DETAILS
  BOUNDED_NUMBER_OF_SESSIONS
  TYPED_MODEL

PROTOCOL
  /home/span/span/testsuite/results/hlpslGenFile.if

GOAL
  As Specified

BACKEND
  CL-AtSe

STATISTICS

  Analysed  : 27 states
  Reachable  : 8 states
  Translation: 0.44 seconds
  Computation: 0.00 seconds
```

Fig. 9. Security simulation results for OFMC and CL-AtSe backends.

Table 3
Symbols and costs in bytes [21].

Symbol	Description	Length (Bytes)
ID	Identification	128
PID	Pseudo identity	32
$H()$	Hash function	32
X	Private key	32
Y, Q	Public key	32
k	Session key	32
σ	Digital signature	32
(J, L)	Blind signature	96
ϕ	Pre-key of session	32
τ	Number of RSUs for fog server	8
ψ	Number of pads for RSU	8
α, v	Seed	20
t	Timestamp	8
VK	Verification key	32
hash chain request	Hash chain request	8
*	Multiplication operator	–
ê	Bilinear Pairing	–
CCS	Authentication Server of the substation	–
RSU	Central Authentication Server	–
HMAC	Hash-based message authentication code	32
P	Point of the elliptical curve	32

Table 4
Communication costs in bytes.

Message	Gunukula et al. [20]	Rabie et al. [21]	Propoced
M1	$32n$	$224n$	$72n$
M2	$128n$	$248n$	$104n$
M3	$168n$	$128n$	$136n$
M4	$136n$	$128n$	$64n$
M5	$32(n*\tau)$	$40(n*\tau)$	$80n$
M6	$32(n*\tau)$	$40(n*\tau)$	$80(n*\tau)$
M7	$32(n*\tau)$	$32(n*\tau*\psi)$	$104(n*\tau)$
M8	$20(n*\tau)$	–	$16(n*\tau)$
M9	$32(n*\tau*\psi)$	–	$32(n*\tau)$
M10	–	–	$32n$
M11	–	–	$32(n*\tau*\psi)$
Total	$n\,(464+\tau\,(116+32\psi))$	$n\,(728+\tau\,(80+32\psi))$	$n\,(488+\tau\,(232+32\psi))$

charging station (EV-RS and RSU-PAT Communications) DSRC communications technology would be used which, within the effective communication range, has better communication performance than LTE. As in [32], the combination of DSRC and LTE has been considered a good solution for VANET".

Communication cost refers to the total number of bytes transmitted by a network during the execution of a protocol. Table 3 shows the values in bytes of each variable used. (Values taken from Rabieh and Wei [21]).

To calculate the communication costs using Table 3 of an EV that will authenticate to the fog server, the first RSU and the first pad, we have:

- $m_1 = \{\phi_{EV}, t_5, H(\phi_{EV}||t_5)\} = 32+8+32 = 72\ Bytes$
- $m_2 = \{\phi_{fs}, VK, t_6, \sigma_{fs}\} = 32+32+8+32 = 104\ Bytes$
- $m_3 = \{PID_1, J_1, L_1, t_7\}_{k_{fs-EV}} = 32+96+8 = 136\ Bytes$
- $m_4 = \{\alpha_1, \tau, PID2_1, t_8\}_{k_{fs-EV}} = 16+8+32+8 = 64\ Bytes$
- $m_5 = \{\alpha_1, \alpha_2, \tau, PID2_1, t_9\}_{k_{C-RSU}} = 16+16+8+32+8 = 80\ Bytes$
- $m_6 = \{PID2_{EV}, 1, t_{10}, HMAC^{\tau}_{RSU}\} = 32+8+8+32 = 80\ Bytes$
- $m_7 = \{H(\alpha_2)^{\tau}, VK_2, t_{11}, HMAC^{\tau}_{EV}\} = 32+32+8+32 = 104\ Bytes$
- $m_8 = \{\text{hash chain request}, t_{12}\}_{k_{rsu-EV}} = 8+8 = 16\ Bytes$
- $m_9 = \{\psi, v, t_{13}\}_{k_{rsu-EV}} = 8+16+8 = 32\ Bytes$
- $m_{10} = \{k_{PH}, t_{13}\}_{K_{c-pat}} = 32+8 = 40\ Bytes$
- $m_{11} = \{H(v)^{\psi}\} = 32\ Bytes$

Table 4 shows the comparison of communication costs between the protocols proposed by Gunukula et al. [20], Rabie et al. [21] and our protocol, counting the bytes (according to Table 3) of the messages exchanged between entity pairs and the total number of messages exchanged by n EVs that try to enter the wireless charging system composed of τ RSUs and ψ pads charging by RSUs.

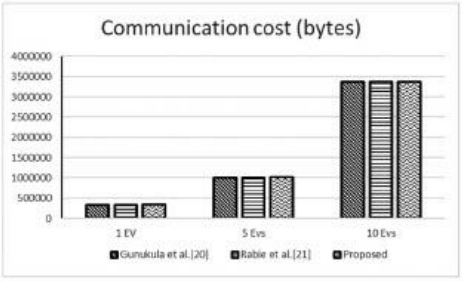


Fig. 10. Communication cost comparisons.

Fig. 10 shows a comparison of the communication costs the protocols proposed in references [20,21] and our protocol. The values adopted for evaluation of computational costs are based on Li et al. [33], who proposed parameters for the modeling of a typical CWD-WPT charging station. According to [33], CWD-WPT is 4,2 km long and the pads are 40 cm long and separated by a 40 cm length. In [20], the RSUs are distributed every 600 m, i.e., 7 RSUs are managed by the fog server and 1500 charging pads are managed by an RSU. It can be verified that the costs of the 3 (three) proposals are very similar, and can be slightly differ, depending on the values of n, τ and ψ, reflecting the structure of CWD-WPT based network.

Table 5
Costs in *ms* of each operation and entity considered (adapted from [34]).

Entity	Parameters of the entities involved			Costs (ms)					
	CPU(GHz)	RAM	OS	T_{mul}	T_{exp}	T_{pair}	T_{hash} [20]	T_{g-sig} [21]	T_{v-sig} [21]
EV/PAD	Qualcomm(R) Octa-core 1,5	2	Android 4,2,2	0,50	0,54	16,6	$0{,}043x10^3$	0,6	0,78
RSUs	Intel(R) Dual-core 3,1	4	64-bit Win-7	0,36	0,38	11,5	$0{,}03x10^3$	0,42	0,55
CCS/CMC/FS	Intel(R) Hexa-core 1,6	16	16 Win server 2012	0,3	0,31	8,6	$0{,}025x10^3$	0,36	0,47

Table 6
Computational costs.

Protocols	EV	CSP-BNK/CMC/FS	RSU	PAD
Gunukula et al. [20]	$2T_{exp}+((\tau+1)^2+(\psi+1)+4)T_{hash}+1T_{v-sig}$	$2nT_{exp}+4nT_{mul}+((\tau+1)n)^2T_{hash}+1nT_{g-sig}+2nT_{pair}$	$(2n+((\tau+1)n)^2T_{hash}$	$n(1-\psi))T_{hash}$
Rabie et al. [21]	$2T_{exp}+(3+\psi)T_{hash}+2T_{v-sig}$	$5nT_{exp}+4nT_{mul}((3+\psi)\tau)nT_{hash}+2nT_{g-sig}+2nT_{pair}$	–	$n(\psi)T_{hash}$
Proposed	$2T_{mul}+((1+\psi)+4)T_{hash}+1T_{v-sig}$	$2nT_{mul}+1nT_{exp}+1nT_{g-sig}+4nT_{hash}+2nT_{pair}$	$4nT_{hash}$	$n(\psi)T_{hash}$

6.2.2. Computational costs

Below is the calculation of the computational costs of the entities of the network model. Table 5 shows the execution times of the Multiplication (T_{mul}), Exponentiation (T_{exp}) and Bilinear Pairing (T_{pair}) functions based on Tao et al. [34], for each entity. The execution costs of the Hash (T_{hash}) function for EV are based on Gunukula et al. [20]. The execution costs for generating a signature message (T_{g-sig}) and its verification (T_{v-sig}) are based on Rabieh et al. [21]. The execution costs of the hash function, signature message and message signature verification for RSU and FS were calculated analytically, taking 70% and 60%, respectively, from the cost of executing these operations to an EV.

The time costs of operations, as symmetric encryption/decryption and addition, have been omitted, because their execution times are very short and rarely used in the protocol, in comparison to the Hash operation.

Table 6 shows a comparison of the number of operations performed by the protocols of Gunukula et al. [20], Rabie et al. [21] and the proposed protocol. Like the other protocols, the proposed protocol performs the operations with higher computational costs in the entity with greater computational capacity (in our case the FS). On the other hand, entities with lower capacity such as EV, RSU, and PADs perform less complex operations to ensure lower latency for the CWD-WPT scheme.

In Fig. 11a comparison of the total computational costs of each entity is shown in the authentication phase of the protocols of Gunukula et al. [20], Rabie et al. [21] and the proposed protocol. The proposed protocol has a better computational cost for EVs, FS

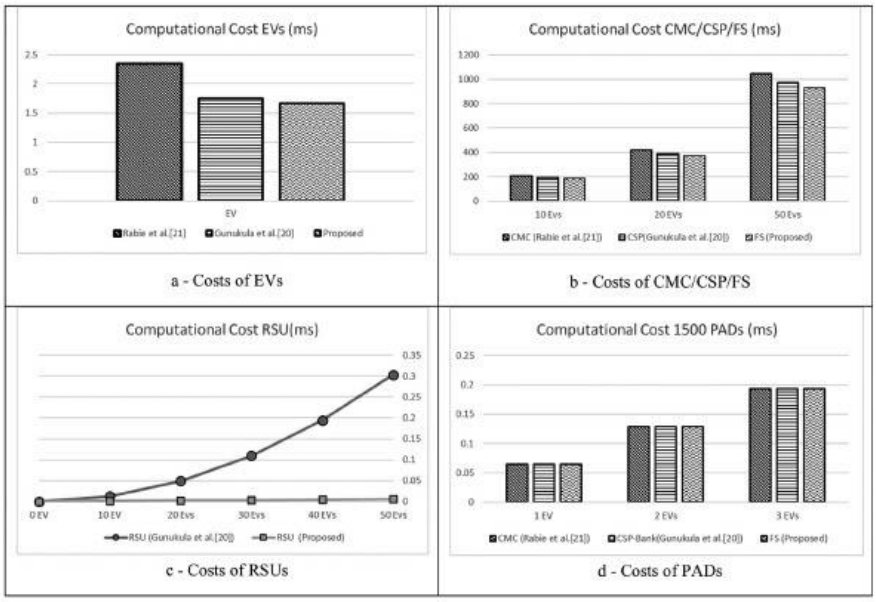


Fig. 11. Computational costs.

and RSU, and maintains the same computational costs of the other protocols for a group of 1500 pads.

7. Conclusions

Communicating things networks (CTN) is the basis for IoT services and therefore must adapt to the particular requirements of each service, such as QoS, data volumes, mobility and interconnection between different devices.

The combination of cloud computing and fog computing in a hierarchical scheme is an effective solution to support next-generation VANETs that are compatible with high mobility, low latency, real-time services and connectivity.

Part of the research related to EVs has been directed at the creation of VANET networks in a cloud environment to support CWD-WTP charging stations. Such stations aim at the optimization and simplification of the charge of EV batteries, since, in this system, cables are not necessary and power is induced while the EV owners drive to their destination.

This work addresses the problems of network security and access control in cloud-based vehicular networks, meeting the most important security requirements such as: authentication, data integrity, confidentiality, access control, non-repudiation and availability. In this sense, this paper aims to contribute for the optimization and security of vehicular networks that support EVs, which has become a trend in several countries due to the global objective to reduce air pollution. The manuscript introduced a new authentication protocol for CWD-WPT charging systems on a VANET network in a cloud and fog computing environment; it is based on digital signatures, HMACs and hashing chains. A short description of some work on authentication in CWD-WPT charging systems has also been provided.

In comparison with other proposals, our scheme has yielded better computational costs and provides better results regarding security analysis and more complete results regarding safety analysis, and avoided problems related to centralization caused by the use of a cloud environment composed of fog computing and cloud computing. Such a combination promotes a better distribution of the computational processing of operations in the devices and guarantees lower latency in communications. Moreover, the protocol has met the security objectives, according to a formal verification conducted by AVISPA tool.

Future work will involve the interaction of EVs in provider, consumer or energy storage modes in CWD-WPT systems, and a simulation of the protocol will be conducted in a network simulator. Another line of work involves authentication and authorization protocols for cyber-physical systems (CPS), and the development of computational trust models for CWD-WPT systems.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] J. Sathishkumar, D.R. Patel, Enhanced location privacy algorithm for wireless sensor network in Internet of Things, in: Proceedings of the International Conference on Internet Things Applications IOTA, 2016, pp. 208–212.

[2] T. Park, N. Abuzainab, W. Saad, Learning how to communicate in the Internet of Things: finite resources and heterogeneity, IEEE Access 4 (2016) 7063–7073.

[3] F.J. Soares, D. Rua, C. Gouveia, B.D. Tavares, A.M. Coelho, J.A.P. Lopes, Electric vehicles charging: management and control strategies, IEEE Veh. Technol. Mag. 13 (1) (2018) 130–139.

[4] X. Mou, O. Groling, H. Sun, Energy-efficient and adaptive design for wireless power transfer in electric vehicles, IEEE Trans. Ind. Electron. 64 (9) (2017) 7250–7260.

[5] T.V. Theodoropoulos, I.G. Damousis, A.J. Amditis, Demand-side management ICT for dynamic wireless EV charging, IEEE Trans. Ind. Electron. 63 (10) (2016) 6623–6630.

[6] Y.J. Jang, Survey of the operation and system study on wireless charging electric vehicle systems, Transp. Res. Part C Emerg. Technol. 95 (2018) 844–866 November 2017.

[7] C. Li, T. Ding, X. Liu, C. Huang, An electric vehicle routing optimization model with hybrid plug-in and wireless charging systems, IEEE Access 6 (2018) 27569–27578.

[8] D. Bavastro, A. Canova, V. Cirimele, F. Freschi, L. Giaccone, P. Guglielmi, M. Repetto, Design of wireless power transmission for a charge while driving system, IEEE Transactions on Magnetics 50 (2) (2014) 2–5.

[9] K. Nahrstedt, H. Li, P. Nguyen, S. Chang, L. Vu, Internet of mobile things: mobility-driven challenges, designs and implementations, in: Proceedings of the IEEE 1st International Conference on Internet-of-Things Design Implementation, IoTDI, 2016, pp. 25–36.

[10] R. Shrestha, R. Bajracharya, S.Y. Nam, Challenges of future VANET and cloud-based approaches, Wirel. Commun. Mob. Comput. 2018 (2018).

[11] M. Ziqian, Z. Guan, Z. Wu, A. Li, Z. Chen, Security enhanced internet of vehicles with cloud-fog-dew computing, ZTE Commun. 15 (S2) (2018) 47–51.

[12] Q.G.K. Safi, S. Luo, C. Wei, L. Pan, Q. Chen, PlaaS: cloud-oriented secure and privacy-conscious parking information as a service using VANETs, Comput. Netw. 124 (2017) 33–45.

[13] C. Huang, R. Lu, K.K.R. Choo, Vehicular fog computing: architecture, use case, and security and forensic challenges, IEEE Commun. Mag. 55 (11) (2017) 105–111.

[14] A. Singh, K. Chatterjee, Cloud security issues and challenges: a survey, J. Netw. Comput. Appl. 79 (2017) 88–115 August 2016.

[15] R. Shrestha, R. Bajracharya, S.Y. Nam, Challenges of future VANET and cloud-based approaches, Wirel. Commun. Mob. Comput. 2018 (2018).

[16] R. Akalu, Privacy, consent and vehicular ad hoc networks (VANETs), Comput. Law Secur. Rev. 34 (1) (2018) 175–179.

[17] R.G. Engoulou, M. Bellaïche, S. Pierre, A. Quintero, VANET security surveys, Comput. Commun. 44 (2014) 1–13.

[18] H. Li, G. Dán, K. Nahrstedt, Proactive key dissemination-based fast authentication for in-motion inductive EV charging, in: Proceedings of the IEEE International Conference on Communications, 2015.

[19] R. Hussain, D. Kim, M. Nogueira, J. Son, A. Tokuta, H. Oh, A new privacy-aware mutual authentication mechanism for charging-on-the-move in online electric vehicles, in: Proceedings of the Eleventh International Conference on Mobile Ad-Hoc and Sensor Networks, MSN 2015, 2016.

[20] S. Gunukula, A.B.T. Sherif, B. Ausby, M. Mahmoud, and X.S. Shen, Efficient scheme for secure and privacy-preserving electric vehicle dynamic charging system, 2017.

[21] K. Rabieh, M. Wei, Efficient and privacy-aware authentication scheme for EVs pre-paid wireless charging services, in: Proceedings of the IEEE International Conference on Communications, 2017.

[22] S. Laporte, G. Coquery, M. Revilloud, V. Deniau, Experimental performance assessment of a dynamic wireless power transfer system for future ev in real driving conditions 1 extended abstract, in: Proceedings of the Ninth International Conference on Future Energy System, e-Energy, 2018, pp. 570–578.

[23] B. Roberts, K. Akkaya, E. Bulut, M. Kisacikoglu, An authentication framework for electric vehicle-to-electric vehicle charging applications, IEEE 14th International Conference on Mobile Ad Hoc and Sensor Systems (MASS), 2017, pp. 565–569.

[24] D. Dolev, A. Yao, On the security of public key protocols, IEEE Trans. Inf. Theory IT-29 (2) (1983) 198–208.

[25] R. Dutta, R. Barua, S. Palash, Pairing-based cryptographic protocols : a survey, in: Proceedings of the IACR Cryptology, 2004, pp. 1–45, ePrint Arch.,.

[26] D. Boneh, B. Lynn, H. Shacham, Short signatures from the Weil pairing, in: Proceedings of the Seventh International Conference on the Theory and Application of Cryptology and Information Security: Advances in Cryptology, ASIACRYPT, 2001.

[27] F. Zhang, K. Kim, ID-based blind signature and ring signature from pairings, in: Advances in Cryptology — ASIACRYPT, 2501, 2002, pp. 533–547, December 2002.

[28] Y. Hu, M. Jakobsson, and A. Perrig, Efficient constructions for one-way hash chains, ICH Q6B, Specifications: Test Procedures and Acceptance Criteria for Biotechnological/Biological Product, no. 1, pp. 1–13, 2001.

[29] J. Kelsey, B. Schneier, D. Wagner, C. Hall, Cryptanalytic attacks on pseudorandom number generators, in: S. Vaudenay (Ed.), Fast Software Encryption - FSE, vol. 1372, Springer, 1998, pp. 168–188.

[30] T. Gunasekhar, K.T. Rao, M.T. Basu, Understanding insider attack problem and scope in cloud, in: Proceedings of the IEEE International Conference on Circuit, Power Computing Technologies ICCPCT, 2015 March 2015.

[31] The AVISPA project; Eeuropean union in the future and emerging technologies (FET open)-AVISPA v1, 1 user manual, 2006, [Online], Available: http://www.avispa-project.org.

[32] Z. Xu, X. Li, X. Zhao, M.H. Zhang, Z. Wang, DSRC versus 4G-LTE for connected vehicle applications: a study on field experiments of vehicular communication performance, J. Adv. Transp. 2017 (2017) 1–10.

[33] H. Li, G. Dan, K. Nahrstedt, Portunes+: privacy-Preserving fast authentication for dynamic electric vehicle charging, IEEE Trans. Smart Grid 8 (5) (2017) 2305–2313.
[34] M. Tao, K. Ota, M. Dong, Z. Qian, AccessAuth: capacity-aware security access authentication in federated-IoT-enabled V2G networks, J. Parallel Distrib. Comput. 118 (2018) 107–117.

Luis Fernando Arias Roman is a Doctorate degree student in Electrical Engineering at the Universidade de Brasília (Brazil) with CAPES scholarship. He holds a Master's degree in Electrical Engineering from Universidade de Brasília (Brazil), a specialization course in Computer Security from Universidad Autonoma de Occidente (Colombia) and also a degree in Electronic and Telecommunications Engineering from Universidad del Cauca(Colombia). He has 8 years of professional experience in areas of communications networks and computer security, and is currently developing research projects in same areas.

Paulo R. L. Gondim obtained a Bachelor's degree in Computer Engineering in 1987, a Master's degree in Systems and Computing from Instituto Militar de Engenharia in 1992 in Rio de Janeiro, and a Doctorate degree in Electrical Engineering from Pontifícia Universidade Católica do Rio de Janeiro in 1998. Since 2003, He has been working at Universidade de Brasília, Brasília, Brazil. He has authored over 70 papers in international periodicals and events and advised 25 master's degree dissertations and 03 doctoral theses. He has been a member of Technical Program Committees of several technical-scientific events and contributed as a reviewer of papers submitted to high-quality periodicals and events. He is a member of the Editorial Advisory Boards of two international journals and has experience in Computer Science, particularly in wireless networking, video streaming, performance evaluation and quality of service/quality of experience assessment. He is a Senior Member of IEEE.

APPENDIX B PUBLICATION IN THE 2021 INTERNATIONAL WIRELESS COMMUNICATIONS AND MOBILE COMPUTING (IWCMC) CONFERENCE.

Authentication protocol built from a chaotic cryptosystem for a fog and cloud-based CWD-WPT charging station

Luis F. A. Roman[a], Paulo de Lira Gondim[a*]
(a) Department of Electrical Engineering, University of Brasilia (UnB), Brasilia, Brazil
*Corresponding Author: pgondim@ene.unb.br

2021 International Wireless Communications and Mobile Computing (IWCMC) | 978-1-7281-8616-0/21/$31.00 ©2021 IEEE | DOI: 10.1109/IWCMC51323.2021.9498756

***Abstract*— In vehicular ad-hoc networks for electric vehicles (EV), the possibility of charging EVs while driving (CWD) with the use of wireless power transfer (WPT) technology offers advantages such as optimization of travel time and greater comfort for travelers. On the other hand, the privacy, confidentiality and integrity of the users' information must be guaranteed for preventing security attacks (e.g., Man-in-the-Middle and Denial-of-service (DoS)). This article presents a new authentication protocol that uses a chaotic map-based cryptosystem for key generation, validation, and key distribution in a cloud-based CWD-WPT loading station. The composite protocol ensures mutual authentication between the recharge system and the EV, as well as privacy and anonymity of users with excellent performance in terms of communication and computational costs, and resistance to attacks.**

***Index Terms*—WPT, CWD, Chaos, Security, Authentication, Fog, VANET.**

I. Introduction

ELECTRIC vehicles are an excellent alternative to reduce fuel consumption and improve environmental conditions in cities. However, some challenges must be overcome (e.g., battery life and user mobility constraints, caused by the long time required for battery charging).

Researchers have bet on a technology called wireless power transfer (WPT) based on magnetic fields created by coils. It induces an alternative current in the vehicle, which is transformed into direct current and stored in the EV battery. The WPT system can reload an EV while it is on the go (or charging while driving - CWD) [1]. The CWD-WPT operation depends on the energy induced in the EV battery by a series of pads embedded in the road pavement.

In the mobility context, a VANET (Vehicular Ad Hoc Network) integrated with the fog and cloud computing capabilities has been considered adequate for solving connection problems, since it is compatible with a CWD-WPT system and constitutes a flexible, safe and efficient system [2]. Such a paradigm enables the design of a hierarchical and scalable architecture that enhances the efficiency of the system distributing the processing and storage of resources among several devices. On the other hand, the treatment of security in cloud and fog-based vehicular networks faces problems caused by heterogeneity, mobility and high vehicle volume.

A special authentication scheme is required for a CWD-WPT system [3], since a CWD-WPT charging station must activate pads for power induction only for system-authenticated and authorized EVs, and EVs and system entities must execute an authentication protocol that ensures the security, confidentiality, and privacy of the system´s user.

Chaos-based encryption has been used for the design of authentication and key agreement protocols, in scenarios such as smart grid and vehicular ad hoc networks, and for security in cloud environments, among other applications. On the other hand, the literature reports authentication protocols for CWD-WPT networks that display security features (e.g., mutual authentication, privacy, anonymity, and integrity), but do not consider some attacks that can affect the system (e.g., injection and Masquerade).

To the best of our knowledge, no study has addressed the use of chaotic maps for ensuring security properties of CWD-WPT systems. This article proposes a chaotic map-based authentication and key agreement protocol for CWD-WPT charging stations, supported by cloud and fog computing. Its operation involves four phases, namely system initialization, EV registration, ticket purchase, and authentication and charging request. The contributions of the study include:

- an authentication and authorization protocol that uses a Chebyshev map-based cryptosystem to ensure privacy, integrity, anonymity, and key distribution;
- a billing scheme based on the creation of blindly signed tickets that avoids compromising the user's private information;
- preservation of the anonymity of EVs, since the protocol is based on tickets purchased offline;
- mutual authentication between the elements of the CWD-WPT charging station and the electrical vehicle;
- an analysis of the security properties and resistance to attacks of the protocol;
- a generalization of the scheme proposed by Tahat et al. [4] for blind signature on multiples tickets; and
- a comparison with other protocols regarding security properties, attacks resisted, and communication and computational costs.

The remainder of the manuscript is organized as follows: Section II discusses some related work; Section III presents the system; Section IV provides preliminary information for a better understanding of the protocol; Section V introduces the protocol; Section VI reports on security and performance analyses; finally, Section VII outlines the conclusions and suggests some future work.

II. Related Work

This section briefly describes some recent proposals of

authentication protocols for CWD-WPT systems.

Revilla et al. [5] developed an anonymous authentication protocol for access control in a CDW-WPT charging station which uses cryptographic primitives, such as ECC-based Diffie-Hellman, exclusive XOR, and modified hash string. The owner of the EV must first buy some tickets in a trusted bank, and then communicate with the service provider. With the help of the bank, the carrier service provider (CSP) validates the currency without exposing the user's identity, and the EV goes to the charging station, which is comprised of road-side units (RSUs) and groups of pads that induce charging to the EV. The protocol resisted attacks such as double spending and man-in-the-middle; however, the authors did not analyze others (e.g., masquerade, impersonation, random number leakage, privileged insider, injection, and DoS).

Li et al. [6] proposed an authentication protocol for a CWD-WPT charging station using hash, AES encryption, digital signatures, and elliptic curve-based subscriptions. The architecture is composed of a CSP and a pad owner located near the wheel that controls the charging pad (CP) installed on the floor of the road to induce energy to the EVs. It guarantees mutual authentication, anonymity and integrity, and resists attacks such as replay and impersonation; however, it does not consider other attacks (e.g., injection, known key, random number leakage, and privileged insider).

III. Network Model and Adversary Model

This section describes the network and adversary models considered in our study. As shown in Figure 1, the CWD-WPT system under study is comprised of:

- a company charging server (CCS), installed on the cloud;
- a set of "τ" RSUs implemented by access points deployed at the margins of the road, with coverage ranging from 300m to 3 km;
- a fog server (FS), installed near the RSUs;
- a group of "ψ" charging pads; and
- electrical vehicles (EVs).

Wireless networks (such as LTE-A – Long Term Evolution – Advanced) enable communication among EVs, FS, and RSUs. The EV<-->pad communication is established through a short-range wireless communication device.

The Dolev-Yao attack (adversary) model [7] is adopted here. It assumes an adversary can reproduce messages, however, its encrypted content cannot be known if this adversary does not have the correct cryptographic keys, and one-way functions are considered unbreakable. In the proposed scheme, only the CCS is trustworthy, hence, safe for storing the identification and banking details of the EV for the purchase of tickets. On the other hand, the FS, the RSU and the pads must not know the identity of the EV or its owner.

IV. Preliminaries on Chebyshev Chaotic Map

Let $\boldsymbol{n}$ be an integer and $\boldsymbol{x}$ a variable within interval [-1,1]. The Chebyshev chaotic maps are defined as

$$\boldsymbol{T_n(x) = cos(n * arccos(x))} \quad (1)$$

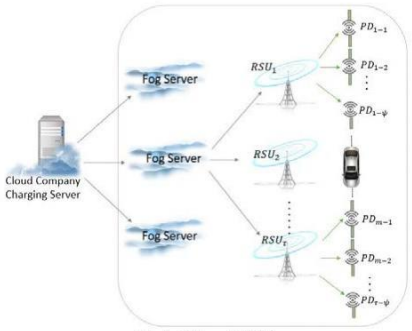


Fig. 1. Network Model

A Chebyshev polynomial establishes recurrence relationships:

$$\boldsymbol{T_0(x) = 1} \quad (2)$$

$$\boldsymbol{T_1(x) = x} \quad (3)$$

$$\boldsymbol{T_{n+1} = 2xT_n(x) - T_{n-1}(x),\ \text{for}\ n \epsilon N} \quad (4)$$

and has the following properties:

- Semi-group property:

$$\boldsymbol{T_r\,(T_s\,(x)) = T_s(T_r(x))} \quad (5)$$

- Chaotic property:

When $n > 1$, the Chebyshev polynomial of degree n, $T_n: [-1,1] \rightarrow [-1,1]$ is a chaotic map with invariant density

$$\boldsymbol{f^*(x) = \frac{1}{\pi\sqrt{1-x^2}}} \quad (6)$$

for Lyapunov exponent $\lambda = \ln(n) > 0$.

- Mathematical problems [8]

According to Zhang [9], in the $[-\infty, +\infty]$ interval, the semigroup property is valid for Chebyshev polynomials.

Therefore, the properties can also be valid for

$$\boldsymbol{Tn(x) = \left(2xT_{n-1}(x) - T_{n-2}(x)\right) mod\ p} \quad (7)$$

where $n \geq 2$, p is a large prime and $x \epsilon (-\infty, +\infty)$. Consequently, as in [10],

$$\boldsymbol{T_r\left(T_s\,(x)\right) \equiv T_{rs}\,(x) \equiv T_s\left(T_r\,(x)\right) mod\ p} \quad (8)$$

- Theorem [3]: if $a = b + c$ and p is a large prime number, then

$$\boldsymbol{(2T_a(M)T_b(M)T_c(M) + 1) mod\ p = ([T_a(M)]^2 + [T_b(M)]^2 + [T_c(M)]^2) mod\ p} \quad (9)$$

Based on the Diffie-Hellman problems assumed difficult to be solved in polynomial time, the Chebyshev polynomial meets the following definitions [8]:

Definition 1. Chaotic maps discrete logarithm problem (CMDLP): Let $\boldsymbol{x}$ and $\boldsymbol{y}$ be two random numbers that belong to the $[-\infty, +\infty]$ range. The calculation of a solution $\boldsymbol{w}$ that satisfies $y = T_w(x)$ is computationally unfeasible.

Definition 2. Computational Chaotic Maps Diffie-Hellman Problem (CCMDHP): Given $x, T_r(x)\ mod\ p$, and $T_s(x)\ mod\ p$, it is infeasible to find $\boldsymbol{r}$ or $\boldsymbol{s}$, from $T_{rs}(x)\ mod\ p$.

V. Proposed Protocol

In our proposal, each FS has a private key x_{fs} and a public key Y_{fs}. RSU also has a private key x_{rsu} , a public key Y_{rsu} ,

and a group key K_{G-RSU} , and is connected to the fog server. On the other hand, the pads are connected to the RSUs and a group key K_{G-pads} is defined for the pads.

1st phase: **Initialization of the System**

Let $\boldsymbol{p}$ be a large prime number and $\boldsymbol{n}$ a factor of $p-1$ and the product of two random prime numbers $\bar{p}$ and $\bar{q}$ ie $n = \bar{p}\bar{q}$. Let β be an element of an infinite group $GF(p)$ of order module $\boldsymbol{n}$ and a generator element of the multiplicative group of set $\boldsymbol{G}$. The system chooses a random number $\boldsymbol{e} \in Z_p^*$ such that $gcd(e,n) = 1$, and a number $\boldsymbol{d}$ such that $e.d = 1\ (mod\ \varphi(n))$, where $\varphi(n) = (\bar{p}-1)(\bar{q}-1)$. The hash function of the system is defined as $H: \{0,1\}^* \rightarrow \mathbb{Z}_p^*$.

The company charging server (CCS) then chooses a master private key $x_{ccs}, \in Z_q^*$ and calculates its global public key $Y_{pub} = T_{x_{ccs}}(\beta) mod(p)$, where $\boldsymbol{T}$ is the Chebyshev polynomial map of degree β defined [2] as a recurrent function:

$$\boldsymbol{T_{x_{css}}(\beta) = (2\rho\, T_{x_{ccs}-1}(\beta) - T_{x_{ccs}-2}(\beta))\ mod(p)} \quad \textbf{(10)}$$

Additionally, it computes its own public key $Q_{ccs} = T_{H(ID_{ccs})}(\beta) mod(p)$ and private key $S_{ccs} = T_{x_{ccs}}(Q_{ccs}) mod(p)$, which are modified in certain periods of time. Finally, the CCS keeps parameters $\{\bar{p}, \bar{q}, d\}$ secret and publishes $\{H, p, \beta, H, Y_{pub}, Q_{ccs}, T\}$.

2nd phase: **EV registration**

When an EV owner decides to use the CWD-WPT charging system service, he/she must register with the CCS through a secure channel. The user then chooses private key $x_{ev} \in Z_p^*$ and calculates public key $Y_{ev} = T_{x_{ev}}(\beta) mod(p)$. The public key along with identity (ID_{EV}) and vehicle charging parameters (VCP), such as battery type, charging level, maximum charging, among other important information for charging, are sent to the CCS to be stored. Finally, the CCS creates a certificate $Cert_{ev} = T_{x_{ccs}}(Q_{ev}) mod(p)$, where $Q_{ev} = T_{H(ID_{ev})}(\beta) mod(p)$, and sends $Cert_{ev}$ and Q_{ev} to the EV.

3rd phase: **Tickets Purchase**

We have assumed messages are exchanged through a secure channel and each ticket is associated with a specific amount of energy to be induced to the EV through a group of pads. The customer buys the tickets, and the money is debited from an associated bank account at CCS. Each user can buy a quantity j of tickets.

The first message, m_1, requesting the purchase of j tickets to the CCS is sent by the EV:

$$\boldsymbol{m_1 = \{J, ID_{ev}, Cert_{ev}\}}$$

The CCS receives it and generates j random values $\{\varepsilon_1, \varepsilon_2, \ldots, \varepsilon_j\} \in Z_n^*$ and $\{r_1, r_2, \ldots, r_j\} \in Z_n^*$, such that each r_i for $1 \le i \le j$ satisfies property $gdc(r_i, n) = 1$, similarly to [4]. For each r_i, $\hat{t}_i = T_{r_i}(\beta) mod(p)$ is calculated and checks if $gcd(\hat{t}_i, n) = 1$. If this is not the case, the EV chooses other values. If the validation is correct, the EV calculates $\alpha_i = T_{\varepsilon_i}(\beta) mod(p)$ for each ε_i, and a message m_2 containing $\Omega = \{\hat{t}_1, \hat{t}_2, \ldots, \hat{t}_j\}$ and $A = \{\alpha_1, \alpha_2, \ldots, \alpha_j\}$ is sent to the EV:

$$\boldsymbol{m_2 = \{\Omega, A\}}$$

Towards ensuring the blindness property of the signature, the EV creates j random values $\{c_1, c_2, \ldots, c_j\} \in Z_n^*$ and j random tuples $\{(u_1, v_1), (u_2, v_2), \ldots, (u_j, v_j)\}$, where u, and $v \in Z_n^*$. Differently from [4], the EV calculates value $\theta_i = c_i \alpha_i$, for all $1 \le i \le j$ and the following functions:

$$\boldsymbol{t_i = T_{u_i+v_i}(\hat{t}_i) mod(p),\ 1 \le i \le j} \quad \textbf{(11)}$$

$$\boldsymbol{\mu_i = u_i^{-1} \theta_i \hat{t}_i t_i^{-1},\ 1 \le i \le j} \quad \textbf{(12)}$$

The EV sends message m_3 with $U = \{\mu_1, \mu_2, \ldots, \mu_j\}$ and $C = \{c_1, c_2, \ldots, c_j\}$ to the CCS.

$$\boldsymbol{m_3 = \{U, C\}}$$

The CCS receives message m_3 with sets U and C, and, similarly to [4]:

$$\boldsymbol{\hat{b}_i = (\mu_i x_{ccs} c_i r_i^{-1} + \hat{t}_i) mod(n),\ 1 \le i \le j} \quad \textbf{(13)}$$

It then sends message $m_4 = \{\hat{b}_1, \hat{b}_2, \ldots, \hat{b}_j\} = \{\hat{B}\}$ to the EV, which calculates:

$$\boldsymbol{b_i = \hat{b}_i^{-e} (\hat{b}_i t_i \hat{t}_i^{-1} u_i + v_i t_i) mod(n), 1 \le i \le j} \quad \textbf{(14)}$$

It then sends message m_5 containing $B = \{b_1, b_2, \ldots, b_j\}$ to the CCS:

$$\boldsymbol{m_5 = \{B\}}$$

After receiving message m_5, CCS calculates:

$$\boldsymbol{\hat{l}_i = (r_i b_i)^d mod(n),\ 1 \le i \le j} \quad \textbf{(15)}$$

and sends the EV message m_6, given by:

$$\boldsymbol{m_6 = \{\hat{L}\},\ where\ \hat{L} = \{\hat{l}_1, \hat{l}_2, \ldots, \hat{l}_j\}.}$$

The EV receives m_6 and calculates:

$$\boldsymbol{o_i = (\hat{l}_i \hat{b}_i) mod(n)} \quad \textbf{(16)}$$

Finally, a valid ticket (θ_i, t_i, o_i) is obtained. Figure 3 shows a summary of the ticket purchase phase.

4th phase: **Charging Request**

This phase describes the verification, authentication, and creation of session keys between the EV and the CWD-WPT charging station.

When the EV owner has a ticket (θ, t, o) and wishes to charge an EV in a CWD-WPT charging station, the EV selects a random number $\sigma_{ev} \in Z_n^*$, calculates $\gamma_{ev} = T_{\sigma_{ev}}(\beta) mod(p)$, and sends an m_1 message to the fog server:

$$\boldsymbol{m_1 = \{\gamma_{ev}, ts_1, H(\gamma_{ev} || ts_1)\}}$$

where ts_1 is a timestamp.

The FS then checks both hash and timestamp. If the verification is successful, it chooses a random value $\sigma_{fs} \in Z_n^*$, and calculates session key $k_{fs-ev} = T_{\sigma_{fs}}(\gamma_{ev}) mod(p)$ and verification key $VK = H(k_{fs-ev})$.

On the other hand, it calculates the following values, so that the EV can calculate the session key and authenticate the FS, respectively:

$$\boldsymbol{\gamma_{fs} = T_{\sigma_{fs}}(\beta) mod(p)} \quad \textbf{(17)}$$

$$\boldsymbol{\eta_{fs} = T_{x_{fs}}(\varpi) mod(p)} \quad \textbf{(18)}$$

where , $\boldsymbol{\varpi = T_{H(\gamma_{fs}, VK, ts_1, ts_2)}(\beta) mod(p).}$ **(19)**

The fog server immediately sends message m_2 to the EV.

$$\boldsymbol{m_2 = \{\gamma_{fs}, VK, ts_2, \eta_{fs}\}}$$

Let us suppose a message m_2' has arrived at the EV, which checks the fog server's signature $\eta_{fs} =? \eta'_{fs} = T_{H(\gamma_{fs}', VK', ts_1, ts_2')}(Y_{fs}) mod(p)$. If the equality is successful, the EV authenticates the fog server, uses the message values to

calculate session key $k_{fs-EV} = T_{\sigma_{ev}}(\gamma_{fs}) mod(p)$, and verifies the integrity of the key calculating $VK = H(k_{fs-ev})$ and checking if $VK' =? VK$. If the equality is successful, the EV uses the session key to crypt and sends fog server message m_3 containing ticket (θ, t, o) and a timestamp.

$$\boldsymbol{m_3 = \{\theta, t, o, ts_3\}_{k_{fs-ev}}}$$

The fog server deciphers the message with session key k_{fs-EV}, the timestamp is checked, and the ticket (θ, t, o) is validated according to the following equation [4]:

$$\left[T_{o^e mod(n)}(\boldsymbol{\beta})\right]^2 + \left[T_{\theta mod(n)}(\boldsymbol{Y_{pub}})\right]^2 + [T_t(t)]^2 = (2T_{o^e}(\boldsymbol{\beta}).T_\theta(\boldsymbol{Y_{pub}}).T_t(t) + 1) mod(p). \quad (20)$$

If the ticket validation is successful, the FS randomly chooses two seeds δ_1 and δ_2 and sends them in an m_4 message along with the number of RSUs τ and a timestamp to the EV.
The message is encrypted with session key k_{fs-ev}.

$$\boldsymbol{m_4 = \{\delta_1, \delta_2, \tau, ts_4\}_{k_{fs-ev}}} \text{ sent to EV}$$

The FS also sends an m_5 message to a group of RSUs (associated with the charging station), which contains the same elements of m_4, but encrypted with RSU group key k_{G-rsu}.

$$\boldsymbol{m_5 = \{\delta_1, \delta_2, \tau, ts_5\}_{k_{G-rsu}}} \text{ sent to RSU}$$

Finally, the fog server revokes ticket (θ, t, o) to prevent its reuse. After receiving m_4, the EV decrypts it and checks its timestamp. If the verification is successful, it calculates a verification key for each RSU using a hash chain [11], $H^{rsu}(\delta_2) = \{H(\delta_2), H^2(\delta_2), \ldots H^\tau(\delta_2)\}$, and, with each verification key, a message authentication code $HMAC_{ev}^{\phi} = \{H^\phi(\delta_1)||d||ts_6|H^\phi(\delta_2)\}$, where ϕ is the position of the RSU at charging station d: $1 \leq \phi \leq \tau$, and authenticates each RSU.

On the other hand, all RSUs receive message m_5 from the fog server, decrypt it with group key (k_{rsu-G}), and check the timestamp. If the check succeeds, each RSU calculates check key $H^\phi(\delta_2)$, a session key $k_{rsu-\delta_1} = T_{x_{rsu}}(T_{H^\phi(\delta_1)}(\beta)) mod(p)$, a verification key $(\boldsymbol{VK})$, and a message authentication code $HMAC_{RSU}^{\phi} = H(H^\phi(\delta_1)||ts_7||H^\phi(\delta_2))$.

The authentication of the EV with an RSU and the group of pads it manages undergo the following authentication process. Initially, when the EV is authenticated with the first RSU, it sends RSU message m_6 containing $H^d(\delta_1)$, the sequence number of RSU, a timestamp, and an $HMAC_{RSU}^1 = H(H^1(\delta_1)||\mathbf{1}||ts_6|H^1(\delta_2))$.

$$\boldsymbol{m_6 = \{H^1(\delta_1), 1, ts_6, HMAC_{rsu}^1\}}$$

When message m_6 arrives, the RSU checks if its database contains $H^1(\delta_1)$. If value $H^1(\delta_1)$ is not found in m_6 or in the RSU database, the communication is terminated. On the other hand, if it is found in the RSU database, the RSU checks $HMAC_{rsu}^1$ with the values associated with $H^1(\delta_2)$. If the verification is successful, the RSU computes session key $k_{rsu-ev} = T_{H^1(\delta_2)}(T_{Y_{rsu}}) mod(p)$, generates a seed λ_1, and finally sends the EV message m_7 containing a value $H^1(\delta_1)$, an RSU pseudo-identification $(PIDrsu = H(ID_{rsu}))$, a key verification code $VK_2 = H(k_{rsu-ev})$, its signature $HMAC_{EV}^1 = H(\mathrm{VK}_2||PID_{rsu}||ts_7||H^1(\delta_2))$, and seed λ_1 along with the number of pads ψ encrypted. It also adds check key $(H^1(\delta_2))$ to a revocation list of RSUs for preventing the reuse of the key.

Additionally, the RSU sends all pads a broadcast message m_8 encrypted with group key (k_{G-pad}) that contains public hash chain [11] verification key $k_{PH} = H^{\psi+1}(\lambda)$ used for the verification of the keys sent by the EV.

$$\boldsymbol{m_7 = \{VK_2, PID_{rsu}, ts_7, HMAC_{EV}^1, \{\psi, \lambda\}_{k_{rsu-ev}}\}}$$

$$\boldsymbol{m_8 = \{k_{PH}, ts_1\}_{K_{G-pad}}}$$

When $m_7' = \{VK_2', ts_7', HMAC_{EV}^{1\,\prime}, \{\psi, \lambda\}'_{k_{rsu-ev}}\}$ arrives, the EV calculates $HMAC_{EV}^1 = H(H^1(\delta_1)||ts_7||H^1(\delta_2))$ and compares it with the $HMAC_{EV}^{1\,\prime}$ that arrived in message m_7 $HMAC_{ev}^{1\,\prime} =? HMAC_{ev}^1$. If $HMAC_{ev}^1$ is valid, the EV authenticates the RSU and uses the message values to calculate session key $k_{rsu-ev} = T_{Y_{rsu}}(T_{H^1(\delta_2)}(\beta)) mod(p)$. It also verifies the integrity of the key calculating $K_2 = H(k_{rsu-ev})$, and compares $VK_2' =? VK_2$. If the equality is successful, it uses the session key to decipher the part of the message that contains seeds $\{\psi, \lambda\}_{k_{rsu-ev}}$.

With values ψ and λ, the EV computes hash chain $H^\psi(\lambda)$. Each group of pads managed by the RSU receives and decrypts broadcast message m_8 with the group key to obtain public hash chain verification key $k_{PH} = H^{\psi+1}(\lambda)$. Whenever a key in a hash chain H^ξ, with $0 \leq \xi \leq \psi + 1$, is sent to one of the pads by the EV in message $m_9 = \{H^\xi\}$, the pad checks if the key has been validated applying hash function $(H^z(H^\xi))$ z times, where $z = \xi - \psi + 1$, and compares it to the public key hash chain (verification key). If the verification is successful, the pad checks the status of the key in the revocation list. If the key has not been revoked, it is accepted and revoked to avoid double use. The process ends when the EV intended (or contracted) load level has been achieved.

VI. Security Analyses

This section reports the security properties and resistance to attacks of our protocol.

Preservation of privacy: during the purchase of tickets by the EV, the CCS keeps the identity of the buyer confidential in the charging phase. FS, RSUs and pads cannot obtain the user's identity from the ticket, and tickets cannot be correlated, since each one is generated from random elements.

Mutual Authentication: the protocol achieves mutual authentication among the EV and the FS and RSUs of the system. The EV authenticates the FS by validating the η_{fs} signature of message m_2. The FS authenticates the valid EV through the Ticket sent in message m3. The RSUs authenticate the EV checking the HMAC that contains element $H^1(\delta_2)$, sent in message m6, and the EV authenticates the RUSs checking the HMC sent in message m7 that contains element $H^1(\delta_1)$.

Integrity protection: the integrity of the messages is maintained by HMACs functions and chaos-based digital signatures. The entities of the system can then detect if an adversary has altered the content of the message.

Perfect Forward Secrecy (PFS), guaranteed as follows:

- random elements, such as σ_{ev}, σ_{fs}, δ_1 , δ_2 are used for the creation of the session key between EV and FS $(k_{(fs-ev)})$ and between EV and RSUs $(k_{(rsu-ev)})$. Even if session keys

$k_{(fs-ev)}$ or $k_{(rsu-ev)}$ are compromised, previous messages cannot be recovered due to the CDH problem;

- during the creation of seed λ between the EV and the Pads, in the worst case, i.e., when λ is compromised, the attacker cannot decipher the previous messages.

Below is an analysis of attacks that affect VANET networks and the resistance of our protocol:

Impersonation: the charging station can analyze whether a ticket is valid or not; however, if an attacker tries to access it, the system detects and expels it. On the other hand, the use of randomly generated session keys for every ticket prevents the attacker from using an old valid key or generating a valid key to access the system.

MitM (Man-in-the-Middle): the use of chaos-based digital signatures and HMACs ensures the integrity of messages and prevents MitM attacks. In the authentication process between the RSU and the EV, only a valid EV can send a correct HMAC with a hash chain generated with the δ_2 seed, which prevents a successful MitM attack.

Replay and Injection: the timestamps and random numbers in messages prevent replay attacks. On the other hand, by applying HMACs and digital signatures functions, the system can detect the injection of data into messages.

Known key: the protocol generates single use tickets and their validity is managed through a revocation list. New session keys are generated with random values for each ticket used by the EV to access the charging station service, which prevents the reuse of tickets or old session keys.

DoS: this attack can be performed in the SF and RSUs. In the former, it is resisted through the validation of the tickets with the public parameters of the system and revocation lists, whereas in RSUs, it is resisted by the efficient validation of connections through HMACs and verification if the authentication variable is in the revocation list. Only a valid user can generate a valid $H(\delta_2)$. Therefore, if an attacker tries to connect to a RSU using a previously used or false $H(\delta_2)$, the RSU rejects communication.

Masquerade attack: the protocol resists Masquerade attacks because the FS and RSUs sign messages with their private keys and random secret elements. Consequently, an attacker cannot represent the messages sent by those entities.

Unlinkability: no entity can link the ticket to a particular EV, since the CCS blindly signs ticket θ_i, and the FS checks it with public values from the system. Table 1 shows a comparison of the security analysis among our protocol and other schemes for authentication.

TABLE 1. COMPARISON OF SECURITY PROPERTIES AND ATTACKS

	[5]	[6]	Proposed Protocol
Mutual authentication	Yes	Yes	Yes
Key agreement	Yes	Yes	Yes
Confidentiality	Yes	Yes	Yes
Integrity	Yes	Yes	Yes
Privacy	Yes	Yes	Yes
Injection attacks	Untreated	Untreated	Yes
Forward secrecy	Untreated	Untreated	Yes
Replay attack	Yes	Yes	Yes
Known key attack	Untreated	Untreated	Yes
DoS attack	Untreated	Untreated	Yes
Man-in-the-Middle attack	Yes	Untreated	Yes
Masquerade attack	Untreated	Untreated	Yes
Impersonation attack	Yes	Yes	Yes
Unlinkability	Yes	Untreated	Yes

TABLE 2. SYMBOLS AND COSTS IN BYTES

Symbol	Description	Length (Bytes)
ID	Identification	128
PID	Pseudo-Identity	32
$H()$	Hash function	32
x, S	Private key	32
Y, Q	Public key	32
k	Session key	32
η	Digital signature	32
(θ, t, o)	Ticket	96
τ	Number of RSUs / FS	8
ψ	Number of pads / RSU	8
δ	Seed	20
ts	Timestamp	8
VK	Verification key	32
p, n, e, d, c	Prime numbers	32
HMAC	Hash-based MAC	32

TABLE 3. COMMUNICATION COSTS IN BYTES

Message	Revilla et al. [5]	Li et al. [6]	Proposed protocol
Total	$n(556 + (\tau 96 + 32\psi)) + 40\tau$	$n(956 + 200\psi)$	$n(440 + (\tau 296 + 32\psi))$

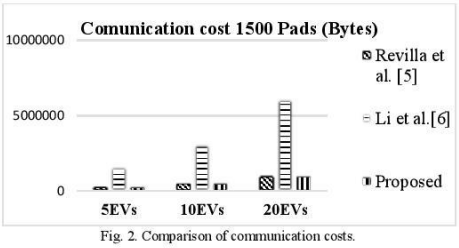


Fig. 2. Comparison of communication costs.

VII. PERFORMANCE ANALYSIS

This section reports on an analysis of the computational and communication costs of the protocol. The authentication processes between FS and EV, EV and RSU, and EV and the pad were assumed independent, since they can be applied at different time periods and locations. A charging station with 1 CCS, 1 FS, 7 RSU and 1500 pads per RSU was considered.

The communication cost is calculated according to the number of bytes of each parameter used in each message, size of each message with respective parameters, and number of messages. Table 2 shows the values in bytes (based on [12]) related to each variable used.

n EVs, τ RSUs and ψ pads (for each RSU) were considered for the communication cost calculation shown in Table 3, thus reflecting the network structure based on CWD-WPT. According to Figure 2, the communication costs of our protocol are lower than those of [6] and very similar to those of [5].

In terms of computational costs, an estimate of the time necessary for the execution of unitary operations that are part of the messages previously described in the phases of the

protocol, as well as the differences among entities regarding the respective processing power are considered. The cost values are based on common and realistic values obtained by experimentation (Tao et al.[13]) and used for performance comparisons of authentication protocols. The methodology adopted considers the cost of each unitary operation multiplied by the number of times each operation is executed, and the messages that include one or more of such unitary operations, as required for the different authentication protocols.

TABLE 4. COMPUTATIONAL COSTS

Protocols	EV	CSP/FS	RSU/ Pad Owner	PAD
Revilla et al. [5]	$5Tmp_{exp}$ + $(3+2\psi)Tmp_{hash}$ $+2Tmp_{pair}$	$(8n+3)Tmp_{exp}$ $+(4n+3)Tmp_{mul}$ $+2nTmp_{pair}$	$2Tmp_{pair}$ $+3nTmp_{hash}$	$2n(\psi)Tmp_{hash}$
Li et al. [6]	$(1+\psi)Tmp_{g-ptns}$ $+2Tmp_{v-ptns}$	$(2n+1)Tmp_{g-ptns}$	$(n)Tmp_{v-ptns}$	$n(\psi)Tmp_{v-ptns}$
Proposed protocol	$3Tmp_{chaos}$ + $((1+\psi)+8)Tmp_{hash}$	$3nTmp_{mul}+2nT_{exp}$ $+10nTmp_{chaos}$ $+3nTmp_{hash}$	$9nT_{hash}$ $+2nTmp_{chaos}$	$n(\psi)T_{hash}$

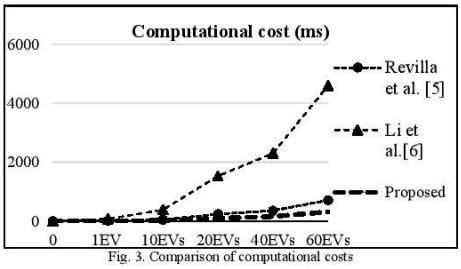


Fig. 3. Comparison of computational costs

Table 4 shows the analytical expression of the computational costs of the different protocols in the authentication phase. Regarding experimentation-based values ([14], [15]) for cost evaluation, Figure 3 shows the superiority of our proposal, in comparison with the protocols of Revilla et al.[5] and Li et al.[6]. Our protocol has a better computational cost for EVs, FS and RSUs due to the use of chaos-based encryption, which has a lower cost compared to elliptical curve encryption and bilinear pairing encryption.

VIII. CONCLUSIONS

A new protocol for authentication and access control based on chaotic cryptography for a CWD-WPT system has been designed. The proposal aims at the optimization of travel times and simplification of battery charge through the induction of energy while EV owners travel to their destination. A comparison with other protocols revealed our scheme achieved excellent results regarding security and performance, and has proven a safe and efficient choice for CWD-WPT systems.

Future research will focus on the security of communications between cloud and fog computing towards support to services of VANET networks, and security of WPT-CWD charging systems with electrical power encryption.

REFERENCES

[1] J. Sathishkumar and D. R. Patel, "Enhanced location privacy algorithm for wireless sensor network in Internet of Things," *2016 Int. Conf. Internet Things Appl. IOTA 2016*, pp. 208–212, 2016.

[2] Y. J. Jang, "Survey of the operation and system study on wireless charging electric vehicle systems," *Transp. Res. Part C Emerg. Technol.*, vol. November 2017, pp. 0–1, 2018.

[3] Y. J. Jang, "Survey of the operation and system study on wireless charging electric vehicle systems," *Transp. Res. Part C Emerg. Technol.*, vol. 95, 2017, pp. 844–866, 2018.

[4] N. Tahat, E.S. et al., "Partially blind signature scheme based on chaotic maps and factoring problems," *Ital. J. Pure Appl. Math.*, no. 39, pp. 165–177, 2018.

[5] M. Pazos-Revilla, A. Alsharif, S. Gunukula, T. N. Guo, M. Mahmoud, and X. Shen, "Secure and Privacy-Preserving Physical-Layer-Assisted Scheme for EV Dynamic Charging System," *IEEE Trans. Veh. Technol.*, 2018.

[6] H. Li, G. Dan, and K. Nahrstedt, "Portunes+: Privacy-Preserving Fast Authentication for Dynamic Electric Vehicle Charging," *IEEE Trans. Smart Grid*, vol. 8, no. 5, pp. 2305–2313, 2017.

[7] D. Dolev and A. Yao, "On the Security of Public Key Protocols," *IEEE Trans. Inf. THEORY*, vol. IT-29, no. 2, pp. 198–208, 1983.

[8] M. Luo, Y. Zhang, M. K. Khan, and D. He, "An efficient chaos-based 2-party key agreement protocol with provable security," *Int. J. Commun. Syst.*, vol. 30, no. 14, pp. 1–9, 2017.

[9] L. Zhang, "Cryptanalysis of the public key encryption based on multiple chaotic systems," *Chaos, Solitons and Fractals*, vol. 37, no. 3, pp. 669–674, 2008.

[10] K. Chain and W. C. Kuo, "A new digital signature scheme based on chaotic maps," *Nonlinear Dyn.*, vol. 74, no. 4, pp. 1003–1012, 2013.

[11] Y. Hu, M. Jakobsson, and A. Perrig, "Efficient Constructions for One-way Hash Chains," *ICH Q6B, Specif. test Proced. Accept. Criteria Biotechnol. Prod.*, no. 1, pp. 1–13,2001.

[12] K. Rabieh and M. Wei, "Efficient and privacy-aware authentication scheme for EVs pre-paid wireless charging services," 2017, doi: 10.1109/ICC.2017.7996868.

[13] M. Tao, K. Ota, M. Dong, and Z. Qian, "AccessAuth: Capacity-aware security access authentication in federated-IoT-enabled V2G networks," *J. Parallel Distrib. Comput.*, vol. 118, pp. 107–117, 2018.

[14] H. Zhu and R. Wang, "A Survey to Design Privacy Preserving Protocol Using Chaos Cryptography," *Int. J. Netw. Secur.*, vol. 20, no. 2, pp. 313–322, 2018.

[15] A. A. Khan, M. Abolhasan, and W. Ni, "5G Next generation VANETs using SDN and Fog Computing Framework," *2018 15th IEEE Annu. Consum. Commun. Netw. Conf.*, pp. 1–6, 2018.

[16] K. Kai, "Fog computing for vehicular Ad-hoc networks : paradigms , scenarios , and issues," *J. China Univ. Posts Telecommun.*, vol. 23, no. 2, pp. 56-65,96, 2016.

APPENDIX C PAPER SUBMITTED TO THE TRANSACTIONS ON EMERGING TELECOMMUNICATIONS TECHNOLOGIES JOURNAL

*

Blockchain and Chaotic Map-based Authentication Protocol for a CWD-WPT Charging System

Luis F. A. Roman, *Member, IEEE* and Paulo R. L. Gondim, *Senior Member, IEEE*

***Abstract*—Although electric vehicles (EV) have become more and more important in the daily transport service, some challenges related to travel time and charging time must still be overcome. A new charging service, called CWD-WPT (Charging While Drive - Wireless Power Transfer), which uses wireless power induction to charge the EV's battery, has been under discussion in recent years. The main idea is to charge the vehicle while it is in motion towards minimizing travel time and maximizing EV autonomy; however, new challenges in access control and information security must be considered. On the other hand, blockchain technology has been recently considered for information security purposes, due to its characteristics such as decentralization and resilience. This article proposes a blockchain-based authentication protocol for a CWD-WPT charging station with centralized control and cloud elements, which, compared with other protocols, has provided excellent safety and performance results.**

***Index Terms*— Blockchain, WPT, CWD, Chaos, Security, Authentication, Fog, VANET.**

I. INTRODUCTION

In recent years, the production of electric vehicles (EVs) has substantially increased due to government incentives, which is aligned with the interests of people in the solution of environmental issues, and been forced to consider the need for renewable energy sources; however, several challenges such as travel autonomy and lack of public charging stations have hampered their wider use.

The literature has recently reported [1][2] the dynamic charging for EVs, named Charging While Drive – CWD, which can be implemented towards overcoming the challenges of autonomy and charging of EVs in public stations. In a dynamic charging station, the EV in motion is charged by magnetic induction induced (or Wireless Power Transfer - WPT) by some pads installed along the highway. A dynamic charging station or CWD-WPT station offers greater advantages to EVs in terms of autonomy and travel time; however, several security challenges related to authenticity, availability, privacy, and integrity of the charging system still must be solved.

An important point for the implementation of CWD-WPT charging stations is the network that will support the service. VANET networks, with a wide variety of applications in intelligent transport systems [3][4] and support to communication between vehicles (V2V) and communication between vehicles and infrastructure (V2I). They are composed of vehicles, Roadside Units (RSUs) located next to the road, and an On Board Unit (OBU) installed in the EVs.

In this manuscript, the VANET architecture considered for the CWD-WPT charging system is based on cloud and fog computing. It is expected to increase the flexibility, scalability, and performance of the services; however, it poses new security challenges to the system [5] [6].

Among the challenges associated with security in cloud- and fog-based VANET networks are data integrity, authenticity, confidentiality, access control, availability, and non-repudiation, which must be solved according to the specific characteristics of the environment related to number of vehicles, mobility, and device heterogeneity[7].

A protocol designed in a cloud and fog computing-based architecture where the charging station is centrally controlled, cryptographic schemes are chaotic maps, and hash chain are used aims at overcoming such challenges. Blockchain has been employed for the creation and management of EV groups and authentication and access control in a CWD-WPT charging station.

The use of chaotic maps offers several advantages for execution performance, enabling a rapid creation of section keys and digital signatures with low computational and storage costs [8][9][10][11]. On the other hand, blockchain provides VANET networks with transparency in their functioning, resistance to attacks, and a quick and efficient validation of the user's credentials in the authentication process for the authorization or denial of access to the system. Moreover, it ensures high service availability due to its decentralized design.

Blockchain has emerged as a decentralized storage mechanism shared by multiple geographically dispersed nodes, but members of a same network. All nodes propagate and check the signed messages transmitted over the network and synchronize the data blocks chained with the use of hash headers created successively with the hash header of the previous block synchronized by a consensus mechanism. Due

*This paragraph of the first footnote will contain the date on which you submitted your paper for review, which is populated by IEEE. It is IEEE style to display support information, including sponsor and financial support acknowledgment, here and not in an acknowledgment section at the end of the article. For example, "This work was supported in part by the U.S. Department of Commerce under Grant BS123456." The name of the corresponding author appears after the financial information, e.g. *(Corresponding author: M. Smith).* Here you may also indicate if authors contributed equally or if there are co-first authors.

The authors are with the Electrical Engineering Department / University of Brasilia (UnB) e-mails: lfroman@aluno.unb.br; ariafernando@gmail.com, . pgondim@ene.unb.br)

APPENDIX D PAPER SUBMITTED TO A JCR-RANKED JOURNAL

Trust Management and Authentication Protocol for CWD-WPT Charging Stations

Luis F. A. Roman[1] and Paulo R. L. Gondim[1]

[1]Electrical Engineering Department – University of Brasília (UnB) - Brasília - Brasil

Abstract: Electric vehicles (EV) have become an important alternative to reduce contamination and atmospheric pollution in the environment caused, in part, by cars, due to their emissions of carbon dioxide. The broad dissemination of EV's in society involves the solution of challenges related to EV charging and travel times, which still must be overcome. Some proposals have pointed to wireless charging while the EVs are driven (CWD) with wireless power transfer (WPT) technology through magnetic induction. However, there are some concerns over security and access control in the system due to the particularities of VANET-based scenario, which requires high performance for offering a quality and safe service. This paper introduces an authentication and access control protocol for a CWD-WPT charging system based on trust management and bilinear pairing. When compared to another one, the protocol shows good performance in terms of computacional, energy, and communication costs. A comparative security analysis performed revealed an improvement by our proposal regarding security functionalities.

Keywords: Authentication, Chaos, EVs, Fog Computing, VANET, CWD-WPT.

1. Introduction

In recent years, electric vehicles have gained importance as a solution for reducing dioxide carbon emissions [1] and several governments have established norms towards offering products that work with clean energies - as an example, Europe has promoted one that will prohibit the production of combustion vehicles from 2035 onwards on renewable energies in the market [2]. On the other hand, several challenges must be overcome before such regulations are imposed, including the charging infrastructure.

Current charging methods consist in plugging EVs into the power grid while parked, which can be uncomfortable for users, since, depending on the charging method and the capacity of the power point where the EV is being charged, it can take several minutes, even hours [3][1]. This has negative consequences on travel time, possible queues caused by charging station demands, and experience of users traveling long distances. One of the solutions to such a challenge is the use of wireless power transfer (WPT), which enables the charging of EV batteries without wires and can be implemented in several ways (e.g., radiowaves (antennas), resonant coupling (resonators), and inductive coupling (coils))[4].

Inductive coupling-based WPT has been used in vehicular networks to charge EV batteries. The system consists of the installation of several charging coils (pads) in a row and their embedding in a lane of the highway (or several lanes) so that an EV with the capacity to collect the energy transmitted by the pads can travel along that highway to charge its battery while in motion (CWD).

The benefits of a CWD-WPT charging station are evident; however, on the other hand, the challenges such a service can offer (e.g., definition of the architecture of the charging station and security and privacy of users of the system) must be analyzed. In this study, the architecture considered is supported by Vehicular Ad Hoc Networks (VANETs) based on cloud and fog