



HAL
open science

Blockchain pour l'Internet des véhicules : une solution IoT décentralisée pour la communication et le paiement des véhicules en utilisant Ethereum

Jabbar Rateb

► **To cite this version:**

Jabbar Rateb. Blockchain pour l'Internet des véhicules : une solution IoT décentralisée pour la communication et le paiement des véhicules en utilisant Ethereum. Intelligence artificielle [cs.AI]. HESAM Université, 2021. Français. NNT : 2021HESAC008 . tel-03609840

HAL Id: tel-03609840

<https://theses.hal.science/tel-03609840>

Submitted on 16 Mar 2022

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

L'École Doctorale Sciences des Métiers de l'Ingénieur
Centre d'études et de recherche en informatique et communications

THÈSE DE DOCTORAT

présentée par : **Rateb Jabbar**
soutenue le : **02 juillet 2021**

pour obtenir le grade de : **Docteur du Conservatoire National des Arts et Métiers**

Spécialité : **Informatique**

**Blockchain for the Internet of Vehicles: A Decentralized IoT
Solution for Vehicles Communication and Payment using
Ethereum**

THÈSE dirigée par

M. BARKAOUI Kamel

*Professeur des Universités, Conservatoire National des Arts
et Métiers, Paris, France*

et co-encadrée par

M. KRICHEN Moez

Professeur Assistant, Université AlBaha, Arabie Saoudite

RAPPORTEURS

M. CHELOUAH Rachid

*Professeur des Universités, École internationale des sciences
du traitement de l'information, Cergy, France*

Mme. NGUYEN Thi-Mai-Trang

Professeur Associé, Sorbonne Université, Paris, France

EXAMINATEURS

M. SECCI Stefano

*Professeur des Universités, Conservatoire National des Arts
et Métiers, Paris, France*

M. SLIMAN Layth

*Professeur des Universités, École d'Ingénieur des Technolo-
gies de l'Information et de la Communication, Villejuif,
France*

Acknowledgements

This Ph.D. dissertation is the result of a three-year research project carried out at the laboratories CEDRIC, CNAM Paris, France and College of Engineering, Qatar University, Doha, Qatar.

The first debt of gratitude is due to my dissertation supervisors Professor Kamel Barkaoui (CEDRIC, CNAM Paris, France) and Dr. Moez Krichen (Albaha University, Saudi Arabia) for their flexibility and freedom provided in formulating my research problems while providing both insightful technical and high-level inputs helped me grow as a scholar. Their guidance together with the trust they showed in my skills are highly appreciated.

I would like to express my sincere gratitude to my advisors Dr. Mohamed Kharbeche and Dr. Noora Fetais for the continuous support and the countless discussions we had throughout my Ph.D. study and research.

Words fail me in expressing my gratitude towards my family. My parents, Adel Jabbar and Saida Turki, inculcated in me the values responsible for making me the person I am today. I thank my sister Rahma, my brother Rouwaid and his wife Mariem for their support and encouragement throughout my graduate studies. I especially wish to thank my fiancé, Olfa, for her enduring love and patience. Without her inspiration and support, I would never have been able to complete this thesis.

Every graduate student needs friends who can keep his sanity in check through this long journey, and I was fortunate to make some. I would cherish the time spent with Ahmed Ben Saïd, Emna Baccour, Mohamed Abdelhedi, Ala Gouisseem and Zina Chkirbene, and their children Line and Youness.

Rateb JABBAR

List of publications

Journal publication

- **Jabbar, R.***, Fetais, N., Shinoy, M., Kharbeche, M., Krichen, M., & Barkaoui, K. “Blockchain for The Internet of Vehicles: How to use Blockchain to secure Vehicle-to-Everything?”, IEEE Sensors Journal (2021).
- **Jabbar, R.***, Kharbeche, M., Al-Khalifa, K., Krichen, M., & Barkaoui, K. (2020). “Blockchain for the internet of vehicles: A decentralized IoT solution for vehicles communication using Ethereum”. Sensors 20.14 (2020): 3928.
- **Jabbar, R.***, Dhib, E., Ben Said, A., Krichen, M., Fetais, N., & Barkaoui, K, (2021) “Blockchain Technology for intelligent transportation system: A Systematic Literature Review”, *Submitted manuscripts with peer review process.*
- **Jabbar, R.***, Krichen, M., Ben Said, A., Abdelhedi, M., Fetais, N., & Barkaoui, K (2021).“Driver Drowsiness Prediction Model Using Machine Learning Techniques for Android Application”, *Submitted manuscripts with peer review process.*

Conference Proceedings Papers

- **Jabbar, R.**, Shinoy, M., Kharbeche, M., Al-Khalifa, K., Krichen, M., & Barkaoui, K. (2020, February). “Driver drowsiness detection model using convolutional neural networks techniques for android application”. In 2020 IEEE International Conference on Informatics, IoT, and Enabling Technologies (ICIOT) (pp. 237-242). IEEE.
- **Jabbar, R.**, Shinoy, M., Kharbeche, M., Al-Khalifa, K., Krichen, M., & Barkaoui, K. (2019, December). “Urban Traffic Monitoring and Modeling System: An IoT Solution for Enhancing

LIST OF PUBLICATIONS

Road Safety”. In 2019 International Conference on Internet of Things, Embedded Systems and Communications (IINTEC) (pp. 13-18). IEEE.

- **Jabbar, R.**, Krichen, M., Kharbeche, M., Fetais, N., & Barkaoui, K. (2020). “A Model-Based Testing Framework for Validating an IoT Solution for Blockchain-Based Vehicles Communication.”
- **Jabbar, R.**, Krichen, M., Fetais, N., & Barkaoui, K. (2020). “Adopting Formal Verification and Model-Based Testing Techniques for Validating a Blockchain-based Healthcare Records Sharing System”. In ICEIS (1), pp. 261-268.
- **Jabbar, R.**, Fetais, N., Krichen, M., & Barkaoui, K. (2020, February). “Blockchain technology for healthcare: Enhancing shared electronic health record interoperability and integrity”. In 2020 IEEE International Conference on Informatics, IoT, and Enabling Technologies (ICIoT), pp. 310-317. IEEE.
- **Jabbar, R.**, Krichen, M., Shinoy, M., Kharbeche, M., Fetais, N., & Barkaoui, K. (2020, June). “A Model-Based and Resource-Aware Testing Framework for Parking System Payment using Blockchain. In 2020 International Wireless Communications and Mobile Computing (IWCMC)”, pp. 1252-1259. IEEE.

* = **Corresponding author**

Résumé

Le concept de villes intelligentes gagne de plus en plus en importance dans les métropoles modernes en raison de l'émergence et de la diffusion d'appareils, de systèmes et de technologies intelligents embarqués et connectés dans la vie quotidienne, qui ont créé l'opportunité de connecter chaque "chose" à Internet. Dans l'ère à venir de l'Internet des objets, l'Internet des véhicules (IoV) jouera un rôle crucial dans la construction d'une ville intelligente. En fait, l'IoV a le potentiel de résoudre efficacement divers problèmes de trafic. Il est essentiel pour améliorer l'utilisation des routes, réduire la consommation d'énergie et la pollution et améliorer la sécurité routière. Néanmoins, le principal problème concernant l'IoV, et en particulier le Véhicule-à-Véhicule (V2V) et le Véhicule-à-infrastructure (V2I), est l'établissement de paiements et de communications sécurisés et instantanés. Pour répondre à ce défi, ce travail propose une solution basée sur la Blockchain pour mettre en place un paiement et une communication sécurisés afin d'étudier l'utilisation de la Blockchain comme middleware entre différents acteurs des systèmes de transport intelligents. Dans cette étude, nous avons évalué les propriétés les plus importantes de la solution développée, à savoir la consommation de la mémoire et de l'énergie, l'immutabilité, la confidentialité, la cohérence, l'intégrité, le temps d'exécution et le coût. L'objet de cette évaluation est de vérifier la capacité de la plateforme basée sur la Blockchain à assurer une communication efficace et un paiement sécurisé avec l'IoV. Selon les résultats, cette plateforme peut contribuer à résoudre les défis les plus critiques de la communication véhicule-à-tout (V2X) en améliorant la sécurité et l'évolutivité.

Mots-clés : Blockchain, Internet des véhicules, Communication automobile, Parking intelligent, Paiements automatisés, Ethereum, Ville intelligente, Système de transport intelligent, Cloud et Android.

Abstract

The concept of smart cities is increasingly gaining prominence in modern metropolises due to the emergence and spread of embedded and connected smart devices, systems, and technologies in everyday lives, which have created an opportunity to connect every “thing” to the Internet. In the upcoming era of the Internet of Things, the Internet of Vehicles (IoV) will play a crucial role in constructing a smart city. In fact, the IoV has the potential to solve various traffic problems effectively. It is critical for enhancing road utilization, reducing energy consumption and pollution, and improving road safety. Nevertheless, the primary issue regarding the IoV, and in particular to Vehicle-to-Vehicle (V2V) and Vehicle-to-Infrastructure (V2I), is establishing secure and instant payments and communications. To respond to this challenge, this work proposes a Blockchain-based solution for establishing secure payment and communication in order to study the use of Blockchain as middle-ware between different participants of intelligent transportation systems. The proposed framework employs Ethereum to develop a solution aimed at facilitating Vehicle-to-Everything (V2X) communications and payments. Moreover, this work qualitatively tests the performance and resilience of the proposed systems against common security attacks. Computational tests showed that the proposed solution solved the main challenges of Vehicle-to-X (V2X) communications, such as security and centralization. In addition, it guaranteed an easy data exchange between different actors of intelligent transportation systems.

Keywords : Blockchain, Internet of Vehicles, Automotive Communication, Smart Parking, Automated Payments, Ethereum, Smart City, Intelligent Transport System, Cloud, and Android.

Contents

Acknowledgements	3
List of publications	4
Résumé	6
Abstract	7
Liste des tableaux	15
Liste des figures	18
Introduction	19
1 Blockchain overview	23
1.1 Introduction	24
1.2 Definition of Blockchain	24
1.2.1 Opportunities and disadvantages	26
1.2.2 Blockchain system types	26
1.3 Blockchain key concepts	27
1.3.1 Asymmetric cryptography	28
1.3.2 Cryptographic Hash Function	29
1.3.3 Hash Pointer	30

CONTENTS

1.3.4	Hash Chain	30
1.3.5	Merkle Tree	31
1.3.6	Transaction	32
1.3.7	Block	33
1.3.8	Peer-to-Peer Network	34
1.4	Blockchain uses cases	34
1.5	History of Blockchain	36
1.5.1	Pre-Bitcoin	36
1.5.2	Blockchain 1.0	37
1.5.2.1	Bitcoin	38
1.5.2.2	Consensus protocols	38
1.5.2.2.1	Proof-of-Work (PoW)	38
1.5.2.2.2	Proof-of-Stake	39
1.5.2.2.3	Delegated Proof of Stake (DPoS)	39
1.5.2.2.4	Practical Byzantine Fault Tolerance (PBFT)	39
1.5.2.2.5	Federated Byzantine Agreement (FBA)	40
1.5.2.2.6	Proof of Authority(PoAu)	40
1.5.2.2.7	Proof of Elapsed Time (PoET)	40
1.5.2.2.8	Proof of Activity (PoAc)	40
1.5.2.2.9	Proof of Burn (PoB)	41
1.5.2.2.10	Proof of Capacity (PoC)	41
1.5.2.3	Mining Incentives	41
1.5.3	Blockchain 2.0	42
1.5.3.1	Smart Contract	42
1.5.3.2	Ethereum	42

CONTENTS

1.5.3.3	Hyperledger	43
1.5.3.4	A comparison between Bitcoin, Ethereum and Hyperledger	45
1.6	Conclusion	46
2	State of the art: Study and discussion of existed related works	47
2.1	Introduction	47
2.2	Internet of Vehicles (IoV)	48
2.2.1	IoV definition	48
2.2.2	The IoV layered architecture models	49
2.3	State of the art literature conferences: classification methodology, statistics and discussion	51
2.3.1	Literature Reviews	55
2.3.2	Security	57
2.3.3	Transports applications	59
2.3.4	Energy	60
2.3.5	Data management	61
2.3.6	Communication and networks	61
2.3.7	Payments and optimization	62
2.4	Open challenges	63
2.5	Conclusion	64
3	Proposed solutions	66
3.1	Introduction	66
3.2	Blockchain IoT Solution for Vehicles communication (DISV)	67
3.2.1	System Overview	67
3.2.2	The Perception Layer	69
3.2.2.1	Android Application for Vehicles (AV)	70
3.2.2.2	Android Application for Infrastructure (AP)	73

CONTENTS

3.2.3	The Network Layer	73
3.2.4	The Application Layer	73
3.2.4.1	Central Cloud Server	74
3.2.4.2	The Blockchain Layer	76
3.2.4.2.1	Blockchain Layer Overview	76
3.2.4.2.2	System In- Depth	76
3.2.5	Nominal Scenario	79
3.2.6	Formal Testing Framework	80
3.2.6.1	Test Generation Principle	80
3.2.6.2	Combining Functional and Load Aspects	81
3.2.6.3	Testing Security Aspects Using Attack Trees	81
3.3	Blockchain IoV Solution for payment (PSEV)	82
3.3.1	The Perception Layer	84
3.3.1.1	Android Auto Application for Vehicles	85
3.3.1.2	Android Application for Parking Space Renters	87
3.3.1.3	Android Application for the Parking IoT System	87
3.3.2	The Network Layer	87
3.3.3	The Application Layer	88
3.3.3.1	Central Cloud Server	88
3.3.3.2	Blockchain sub-layer	88
3.3.3.2.1	Access Management System	89
3.3.3.2.2	Parking Payment Management System	90
3.3.3.2.3	Communication Management System	90
3.3.4	Nominal Scenarios	90
3.3.4.1	Parking Management System	90

CONTENTS

3.3.4.1.1	Requesting for Parking	90
3.3.4.1.2	Providing parking as a supplier	92
3.3.4.2	Communication Management System	92
3.3.5	Formal Testing Framework	93
3.3.5.1	Model-Based Testing	93
3.3.5.2	Combining Functional and Load Aspects	95
3.3.5.3	Improving Formal Testing Methods	95
3.3.5.4	Testing Dynamic Adaptations	97
3.3.5.5	Test Isolation	98
3.3.5.6	Optimization of Testers Placement	98
3.3.5.6.1	Different Types of Constraints	98
3.3.5.6.2	Objective Functions	99
3.3.5.6.3	Algorithms	99
3.4	Conclusion	100
4	Performance evaluation	101
4.1	Introduction	102
4.2	Blockchain IoV Solution for Vehicles communication (DISV)	102
4.2.1	Costs	103
4.2.2	Execution Time	104
4.2.3	Memory and Power Consumption	106
4.2.4	Availability	107
4.2.5	Integrity	108
4.2.6	Consistency	109
4.2.7	Confidentiality	110
4.2.8	Immutability	110

CONTENTS

4.2.9	Security	111
4.3	Blockchain IoV Solution for payment (PSEV)	113
4.3.1	Cost	114
4.3.2	Execution Time	115
4.3.3	Integrity	117
4.3.4	Consistency	118
4.3.5	Confidentiality	119
4.3.6	Immutability	119
4.3.7	Memory and Power Consumption	120
4.3.8	Security	122
4.3.8.1	Mythril and MythX	122
4.3.8.2	OWASP	122
4.3.8.3	SCSVS	123
4.4	Discussion	127
4.4.1	Cost	127
4.4.2	Execution Time	127
4.4.3	Memory and Power Consumption	128
4.5	Conclusion	128
	Conclusion	130
	Extended Summary in French	135
	Glossary of acronyms	163

List of Tables

1.1	Features comparison of the well-known Blockchain technologies: Bitcoin, Ethereum, and Hyperledger	45
2.1	First five most cited research papers	55
2.2	The exhaustive list of current literature surveys on Blockchain technology	56
2.3	Research classification projected on research directions and IoV layer correspondence.	59
3.1	The main features of the developed IoT solution.	69
3.2	Accuracy per driving using CNN method	72
3.3	Model of the sent message in the Blockchain network.	77
3.4	Urgency level of the message sent through the Blockchain Network.	78
3.5	Type of message sent through the Blockchain Network.	79
3.6	The main features of the developed PSEV solution.	83
4.1	Costs of the different functions in the Smart Contract based on 1 ETH = 161,92 USD and 1 gas = 0,000000001 ETH Rates.	104
4.2	Open Web Application Security Project (OWASP) top web vulnerabilities and security and privacy requirements.	112
4.3	Ethereum top web vulnerabilities and Security and privacy requirements.	113
4.4	Functions cost of the Smart Contract based on the used Rates.	115
4.5	Execution time of most called functions of the Smart Contract in milliseconds.	117

LIST OF TABLES

4.6	OWASP top web vulnerabilities and security and privacy requirements.	124
4.7	Ethereum top vulnerabilities and Security requirements.	125
S1	Le taux du coûts des différentes fonctions du contrat intelligent basé sur 1 ETH = 161,92 USD et 1 gas = 0,000000001 ETH	150
S2	Les coûts des fonction de contrat intelligent.	155
S3	Temps d'exécution de la plupart des fonctions demandées du contrat intelligent en millisecondes.	156

List of Figures

1.1	Key concepts of Blockchain	26
1.2	Generic Chain of Blocks	27
1.3	Asymmetric Cryptography procedure	28
1.4	Hashing based cryptography procedure	29
1.5	Hash pointer illustration	30
1.6	Hash Chain	31
1.7	Hash Merkle Tree	32
1.8	Block	33
1.9	Blockchain Applications in Different Domains	34
1.10	History of Blockchain technology	37
2.1	Current IoV layered architecture models	49
2.2	Number of published papers per Year	52
2.3	Number of papers by country	53
2.4	Documents by affiliation	54
2.5	Blockchain for transportation	54
3.1	The architecture of the proposed Internet of Things solution.	69
3.2	Video preprocessing outline	70
3.3	Facial feature detection	71

LIST OF FIGURES

3.4	Screenshot of the four main pages of the Android application for Vehicles (AV).	73
3.5	Screenshot of a real trip displaying information about the vehicle and the driver. . . .	74
3.6	Screenshot of a real trip displaying the data recorded for every event.	75
3.7	Screenshot of a real trip displaying the Blockchain layer for every road section.	75
3.8	The architecture of decentralized applications (Dapp).	76
3.9	Sequence diagram of nominal scenario of communication between Internet of things (IoT) devices.	80
3.10	Test generation principle	81
3.11	An example showing how the response time of the system under test varies regarding the current load level.	81
3.12	Example of an Attack Tree	82
3.13	The developed Internet of Vehicle architecture PSEV.	84
3.14	Screenshot of the Android application interface inside the vehicle.	85
3.15	Screenshot of the primary navigation pages of the Android application.	86
3.16	Screenshot of the web portal of the central cloud.	89
3.17	Screenshot of Android Application for Vehicles displaying requesting for parking. . . .	91
3.18	Screenshot of developed web application.	92
3.19	Screenshot of a message received to alert the driver.	93
3.20	A Timed Automata with 5 states, 4 transitions, 3 actions and 1 clock.	94
3.21	Test generation principle	95
3.22	Test generation procedure	95
3.23	A System Under Test which has three possible behaviors corresponding to different load levels.	96
4.1	System evaluation diagram.	103
4.2	Execution time of the different functions in the Decentralized IoT solution for Vehicles communication (DISV) in milliseconds.	105

LIST OF FIGURES

4.3	Comparing Memory Consumption of DISV with commercial mobile applications. . . .	107
4.4	Comparing Energy Consumption of DISV with commercial mobile applications. . . .	107
4.5	Execution time of most called functions of PSEV-Communication.	116
4.6	Comparison of memory consumption of PSEV and mobile applications commercially available.	120
4.7	Comparing Energy Consumption of PSEV with commercial mobile applications. . . .	121
S1	L'architecture de la solution IdO proposée.	138
S2	Capture d'écran des 4 pages de l'application Android pour les Véhicules (AV).	140
S3	L'architecture Internet des véhicules développée PSEV.	143
S4	Capture d'écran de l'interface de l'application Android à l'intérieur du véhicule.	144
S5	Capture d'écran des principales pages de navigation de l'application Android.	145
S6	Temps d'exécution des différentes fonctions de la solution IdO décentralisée pour la communication des véhicules (DISV) en millisecondes.	151
S7	La comparaison de la consommation de mémoire du DISV avec des applications mobiles commerciales.	152
S8	La comparaison de la consommation d'énergie du DISV avec des applications mobiles commerciales.	153
S9	Temps d'exécution des fonctions les plus demandées de PSEV-Communication.	157
S10	La comparaison de la consommation de mémoire du PSEV avec des applications mobiles commerciales.	158
S11	La comparaison de la consommation d'énergie du PSEV avec des applications mobiles commerciales.	159

Introduction

With a rapid increase in the number of vehicles over the last two decades, and in spite of notable improvements in the infrastructure, the transportation solutions that were in place have become insufficient to handle today's ever-increasing traffic growth. The need to integrate Intelligent Transportation Systems (ITS) has become more critical. In particular, the purpose of the ITS is to reduce traffic problems, enhance traffic efficiency, and contribute to the development of smart roads. Users receive valuable information about seat availability and other traffic conditions. Accordingly, safety and comfort are increased, whereas commuting time is reduced. Owing to the rapid development of innovative computation and communication technologies, the original concept of Vehicular Ad-hoc networks (VANETs) was transformed into an innovative concept entitled the Internet of Vehicles (IoV) [1–3]. The IoV is a necessary pre-requirement of the ITS, as it enables the interconnection of smart vehicles on the internet. According to the US Department of Transport (DOT) [4], the IoV will particularly contribute to the reduction of crashes that include unimpaired drivers. More precisely, the use of the IoV can avoid 79% of such crashes because it allows effective communication and collaboration among vehicles. Moreover, it also includes communication with bicycles, pedestrians, and roadside infrastructure. By exchanging messages about traffic conditions and information about safety and accidents, global traffic control can be achieved with the purpose of reducing environmental pollution, accident rates, and traffic jams [5, 6] while enhancing convenience, comfort, and safety. Consequently, also public transportation and pedestrian traffic can be significantly improved.

A rapid rise in the ITS is expected in the next years, through initiatives such as ERTICO - ITS Europe [7] and CityVerve Manchester [8], which will, in turn, contribute to the development of smart cities. The IoV ensures the interconnection among smart vehicles, roadside infrastructures, and pedestrians to respond to the increasingly complex functional requirements of the ITS and enable the vehicle-to-everything (V2X) paradigm. However, a growing number of smart vehicles and related vehicular applications and services will inevitably create enormous amounts of data and network traffic. Moreover, the complex characteristics and context of the IoV, low latency, and high mobility will bring challenges related to security, management and cloud-based storage. Furthermore, it is necessary to ensure the compatibility and interoperability of IoV entities using different service providers. Therefore, it is paramount to ensure the storage and data exchange IoV platform that is scalable, flexible, interoperable, distributed, and decentralized to ensure the further development of the IoV and realization of the full potentials of the ITS.

Furthermore, Blockchain technology [9] has fundamentally transformed digital currencies by introducing Bitcoin [10]. Blockchain represents a distributed ledger that can maintain an immutable log of transactions occurring within a network. Although primarily the research focus was on the use of Blockchain in the financial sector, recently the scientists have shifted attention to the Internet of Things (IoT) [11] and have started using it to generate a truly decentralized, trustless, and secure environment. The development of Blockchain has enabled the emergence of high technology in the sensitive and active sectors by allowing reliability of information via consensus, the immutability of records, and transaction transparency. However, the most important achievement of Blockchain is enhanced security and trust. In addition, owing to smart contracts, the optimization and automatization of the information handling process and saving of costs has been achieved. Blockchain technology has numerous advantages in comparison to traditional centralized architectures. Yet, disadvantages such as storage limitation, flexibility, security, and high costs, should be noted and taken into consideration.

The combination of Blockchain technology with IoV brings considerable benefits and opportunities. More precisely, the integration of Blockchain technology and the IoV can significantly improve the security, intelligence, big data storage, and efficient management of the IoV.

This Thesis proposes a Blockchain-based solutions for establishing secure payment and communication in order to study the use of Blockchain as middle-ware between different participants of intelligent transportation systems. The proposed framework employs Ethereum to develop a solution aimed at facilitating Vehicle-to-Everything (V2X) communications and payments.

The thesis is divided into 4 chapters . A brief overview of the following chapters is provided below:

- Chapter 1: *Blockchain overview*: This chapter outlines the chronological development of the Blockchain technology from the Pre-Bitcoin phase, characterized by the fundamental cryptographic systems, to the Blockchain 2.0 phase, characterized by Hyperledger and Ethereum implementations and smart contracts.

- Chapter 2: *State of the art*: This chapter provides a comprehensive literature review of the intelligent Transport applications for the IoV networks and classifies the relevant studies into six main areas: Security, Transport applications, Energy, Communication and network, Data management, and Payments and optimization. Furthermore, we classified the current contributions for each area on the basis of the IoV layers.

- Chapter 3: *Proposed solutions*: This chapter discusses the proposed IoV solutions with Real-Time Application (RTA) aspect aimed at establishing secure communications and payments in the transportation systems. The architecture of proposed solutions is consists of the perception, network, and application layers. Furthermore, the detailed information of nominal scenarios cases for utilizing this solution are presented.

- Chapter 4: *Performance evaluation*: This chapter assesses the specific properties required for the straightforward functioning of the solution. The main properties of the proposed solution are execution time, costs, availability, integrity, immutability, and security. All these properties must achieve the highest performance in order to enable the flawless operations of the solution.

Chapter 1

Blockchain overview

Contents

1.1	Introduction	24
1.2	Definition of Blockchain	24
1.2.1	Opportunities and disadvantages	26
1.2.2	Blockchain system types	26
1.3	Blockchain key concepts	27
1.3.1	Asymmetric cryptography	28
1.3.2	Cryptographic Hash Function	29
1.3.3	Hash Pointer	30
1.3.4	Hash Chain	30
1.3.5	Merkle Tree	31
1.3.6	Transaction	32
1.3.7	Block	33
1.3.8	Peer-to-Peer Network	34
1.4	Blockchain uses cases	34
1.5	History of Blockchain	36
1.5.1	Pre-Bitcoin	36
1.5.2	Blockchain 1.0	37
1.5.3	Blockchain 2.0	42
1.6	Conclusion	46

1.1 Introduction

The revolution power of the Blockchain has been recognized worldwide, leading to a high priority technology for most active and sensitive sectors. Transparency in transactions, immutability of records, reliability of information via consensus, and improvement of security and trust over a system present the most innovative benefits of the Blockchain technology. Furthermore, smart contracts brought a new view to establish contracts and financial insurances helping to optimize and automate the information handling process and saving relative cost. Intelligent Transport Systems (ITS) have also benefited from this revolution. In this work, we study the contribution of Blockchain in ITS and more particularly in IoV networks. The main contributions could be summarized as follows: we discuss the chronological Blockchain evolution from the classic crypto-currencies systems to the last Hyperledger implementation. Interested in the role of Blockchain in IoV networks, we recognize the first contributor to classify and analyse the related literature, published from January 2015 to July 2020, according to their research sub-axis and their belonging to the appropriate IoV layer. The rest of the chapter is structured as follows: Section 1 presents a global overview of the Blockchain technology, as well as its opportunities, disadvantages, related to cryptography fundamentals, relative taxonomies, and different uses cases. Besides, it introduces a detailed chronological evolution of the history of Blockchain, divided into three main phases namely the Pre-Bitcoin, Blockchain1.0, and Blockchain 2.0. It exposes also the existed consensus algorithms used in the literature, introduces and compares the three popular Blockchain implementations: Bitcoin, Ethereum, and Hyperledger.

1.2 Definition of Blockchain

Blockchain is an innovative technology that forms the basis of crypto-currency Bitcoin [12] created by Satoshi Nakamoto (pseudonym) who proposed a contribution on Bitcoin in 2008 and was released in 2009. However, the original paper did not discuss the Blockchain [13]. Therefore, it is possible that Blockchain is an unintentionally invented technology with a potential to be applied in numerous fields. The purpose of Blockchain is to ensure that transactions of value within a network of untrusted entities go through a trusted intermediary [14]. The emergence of Blockchain is contributing to a paradigm shift in computer science. The aim of this section is to explain the concept of Blockchain illustrated in Figure 1.1 , its beginnings, its development and its relevance for the proposed solution.

1.2. DEFINITION OF BLOCKCHAIN

A Blockchain represents a database structured as a one-dimensional hash chain of blocks whose origins are in a genesis block. The distribution and the maintenance of a Blockchain are done by a set of participants of a peer-to-peer network that do not trust each other. Thus, there must be a consensus mechanism among the participants, so that they all can agree about the state of the database. The introduction of a data structure to fingerprint the data would enhance a Blockchain storage efficacy. In particular, digital signatures should be used to ensure that the adequate identity issues the changes to data. Essentially, a Blockchain is defined as "a one-dimensional hash chain". It is argued that a Blockchain is a linked list implemented with hash pointers [15]. Nevertheless, some scholars, such as Abeyratne and Monfared [16], argue that a Blockchain is a database distributed in a peer-to-peer network. Nevertheless, the distributed property is not a requirement for a Blockchain, but rather a neat application of the Blockchain database. The Blockchain is powerful due to this property and, accordingly, it is often known as a distributed database. A decentralized distribution means that the participants do not need to trust each other in order to manage a Blockchain [17]. However, a distributed database requires a consensus mechanism so that the same version is ensured on all sites. The database depends on a chain structure. Accordingly, long chains consume the considerable memory. A hash pointer stored in one block points to the previous block. Hence, it is not possible to modify data in the previous block without invalidating the pointer in the next block. It means that the Blockchain is invalidated by removing unnecessary data. The Merkle tree [13] can resolve this challenge, as the participants would be able to keep only a valid copy of the data relevant to them by fingerprinting the transactions by using a data structure. The use of data structure for fingerprinting the data is not necessary for a Blockchain. However, it is a great tool to increase the storage efficiency for the Blockchain participants and accordingly, the overall usefulness of the Blockchain. Using digital signatures in a Blockchain ensures the origin of issued database transactions. In this way, data can be linked to the owner. Yet, this tool is not a requirement for the Blockchain. For a transaction to be valid, it must be consistent. It means the adequate identity corresponding must issue transactions to the altered data, which can be achieved by using digital signatures. Concerning the monetary system, it indicate spending only one's own money.



Figure 1.1: Key concepts of Blockchain

1.2.1 Opportunities and disadvantages

Blockchain is now a wide range technology beyond cryptocurrency applications. Its decentralized architecture relieves the system from known central authority limits such as security vulnerabilities, bottlenecks access to networks, etc. Historical and current data and relative changes are all transparently recorded and publicly viewable to any seeker. Blockchain is also an immutable ledger where data are difficult to alter or to tamper with. Besides, it ensures a high and secure data sharing channel owing to the efficient decentralized cryptographic mechanisms that it employs. Moreover, data exchanging operations and transactions are faster and cost effective compared to traditional systems. However, Blockchain still poses inefficiency to deal with big data. The two famous Blockchain implementations, Bitcoin and Ethereum, have 200 Go and more than 1 TB size respectively. It is also essential to involve new and well-rewarded participants to maintain security, efficiency and widespread usage of Blockchain.

1.2.2 Blockchain system types

Three primary genres of Blockchain systems are commonly discussed in literature [18–22]. Private Blockchain, also called permissioned Blockchain, is a closed and access restricted system where only persons meeting certain requirements and pre-verified ones are allowed to perform certain actions on it. Organisations, mostly in small range, and business Blockchain reserve this Blockchain genre, however, it is not suitable for trading scenarios. In the other hand, public Blockchain presents a wide open, freely-joining permissionless system where anyone can join and have full rights to use it. Auditability and transparency of information are more appreciated since no access limitation is imposed. But,

1.3. BLOCKCHAIN KEY CONCEPTS

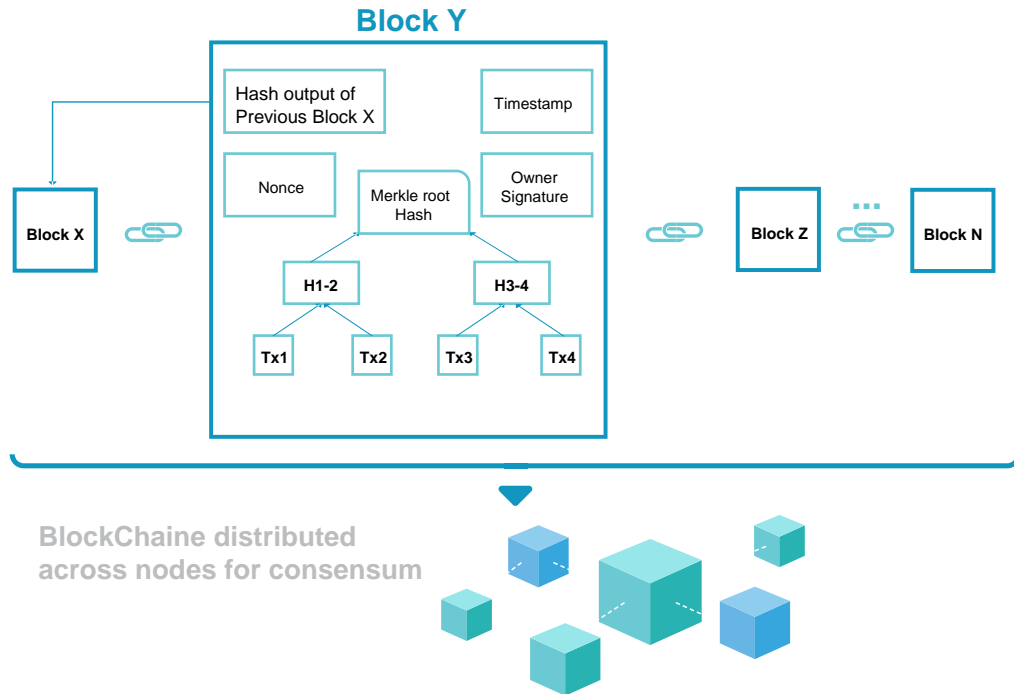


Figure 1.2: Generic Chain of Blocks

delay and cost of related mining operations and synchronization among all participant nodes are high. Public Blockchain are not recommended for high delay or energy sensitive domains. While these two Blockchain genres are pretty self-explanatory, the third one is in between: consortium Blockchain is a semi-private and semi-open Blockchain system where only organizations or participants with same goals could join the group. It ensures scalability, acceptable delay and reasonable costs.

1.3 Blockchain key concepts

This section describes basics concepts of cryptography related to key pair encryption, hashing function, Merkle Tree and defines Blockchain taxonomy such as data block and transactions presented in Figure 1.2.

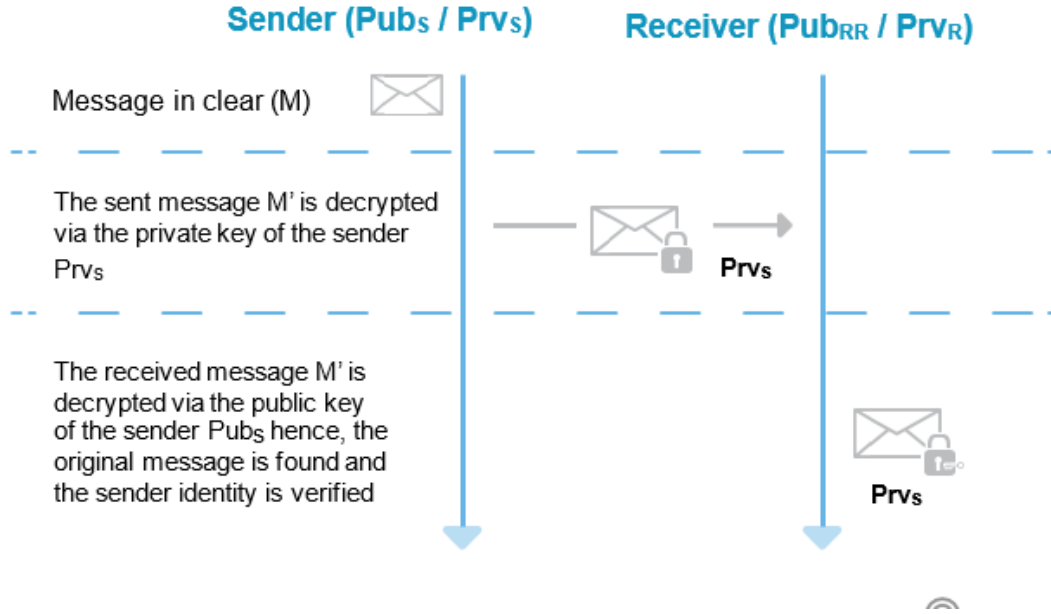


Figure 1.3: Asymmetric Cryptography procedure

1.3.1 Asymmetric cryptography

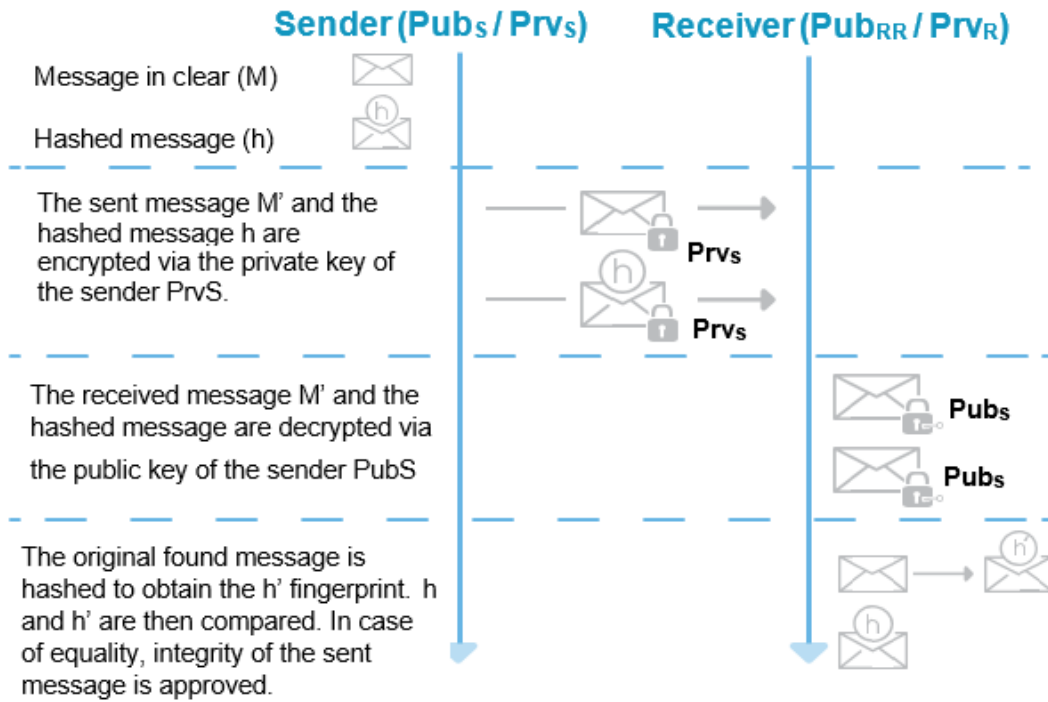
Asymmetric Cryptography (AC), also known as public key cryptography, is a technique of encryption and decryption of data that offers a very high level of security, information protection, authenticity and confidentiality of a data. In Blockchain domain, asymmetric cryptography allows a user to sign a transaction made on the public register of the Blockchain and therefore to certify that he is the author. Based on a key pair [22], each user possesses his proper pair of key (Public key / Private key). The private key remains private to its belonged user, while the public key is accessible by others participants. Let M denotes the clear transferred message and M' the sent message as shown in Figure 1.3. M' is always encrypted or signed by the sender's private key as shown in Equation 1.1:

$$M' = \text{Sign}(\text{PrivateKey}(\text{PrvS}); \text{Message}(M)) \quad (1.1)$$

In turn, the receiver checks which public key decrypts the encrypted message, and then, the sender identity is disclosed [23]. Equation 1.2 illustrates the decryption or the verification phase:

$$M = \text{Verify}(\text{PublicKey}(\text{PrvS}); \text{Message}(M')) \quad (1.2)$$

According to asymmetric cryptography philosophy, the reliability of the authentication is confirmed



Prv_S: Private key of the sender S ; Pub_S: public key of the sender S,
 Prv_R: Private key of the receiver R ; Pub_R: public key of the receiver R

Figure 1.4: Hashing based cryptography procedure

considering that the signature is bound to a signer. Besides, the provision of non-repudiation is not sufficient to allow a sender to deny sending a particular message. Moreover, the message cannot be changed while in transit, as it would imply invalidation of the verification. Thus, integrity is ensured. Moreover, digital signatures guarantee a message origin because the senders cannot forge a signature due to the previously signed messages [24].

1.3.2 Cryptographic Hash Function

Figure 1.4 illustrates the hashing based cryptography scenario. In fact, hash functions are widely used in Asymmetric Cryptography for integrity verification purposes. A cryptographic hash function, noted H, is an algorithm that accepts variable input length message, noted M and outputs a fixed length digest or fingerprint, noted h [20]. Equation 1.3 formulates mathematically the hash function. It is also known as a one-way hash function since it is nearly impossible to recover the original message M from the fingerprint h. Additionally, it is hard to find two different messages inputs M and N that



Figure 1.5: Hash pointer illustration

hash to identical output 1.4, for that, the hash function is called collision free. If collisions occur, it would be very rarely [25].

$$h = H(M), M : message \tag{1.3}$$

$$H(M) \neq H(N), if M \neq N, M, N : messages \tag{1.4}$$

MD5, SHA-1, SHA-3, SHA-256, SHA-512 are examples of commonly used hash functions. The last three ones are used to ensure authenticity for Blockchain.

1.3.3 Hash Pointer

The hash pointer function, illustrated by Figure 1.5, represents at the same time the cryptographic hash of some data and a pointer to the storage location. The input can be calculated with the pointer and data are anti-tamper guaranteed [15]. The integrity property is derived from the cryptographic hash function: consider two messages M and N, their hash digests, H(M) and H(N), are equal if and only if messages are identical, M = N. Thus, the content changes with the change in the hash value of a data element.

1.3.4 Hash Chain

When hash pointers are used to link different data elements, the process is known as a hash chain [20]. A hash pointer represents the head of the chain. A one-dimensional hash chain represents a linked list appropriate hash pointer derived from a genesis element as presented in Figure 1.6. An

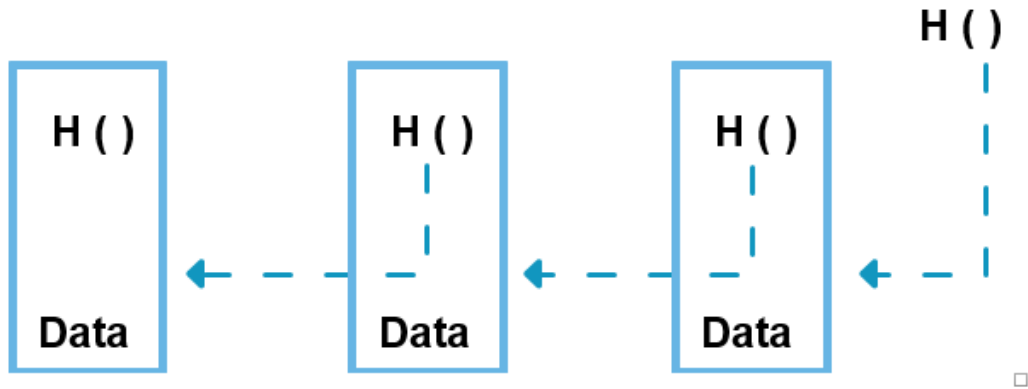


Figure 1.6: Hash Chain

alteration of one element invalidates all subsequent elements of the chain. Hence, a hash pointer update is highly recommended. The hash function is collision free. Consequently, the value of the hash pointer in the next element and the hash of the tampered element cannot be the same. Moreover, it is possible to detect if the chain was tampered. In fact, when one element is tampered, all subsequent elements must be changed as well to preserve the chain consistency. It also modifies the head of the chain, which compromises the integrity of the entire chain [15].

1.3.5 Merkle Tree

The Merkle tree, also called Hash tree, presents a binary data structure used to encode more securely and efficiently Blockchain data. Instead of hashing an entire large data block, which is a time costly operation, the block is partitioned into small data elements. In turn, each element is hashed separately. The correspondent hash digests are grouped by pair, concatenated and re-hashed until completing all the data elements of the current block. In some cases, a block contains an odd number of data elements. To do, one element is then doubled before executing the hash. Figure 1.7 illustrates a highly simplified data block with only four data elements. The bottom layer of the tree represents the hash digest relative to each data element. Hence, “Hash1” is the hash print of “Data1” and so on. This layer refers to the leaves nodes of the Merkle Tree. The intermediates hashes form the branches nodes which are the hash of its respective child nodes (leaves or branches nodes). The top hash presents the root. The Merkle proof [26] allows verification of a leaf value by comparing the public Merkle root

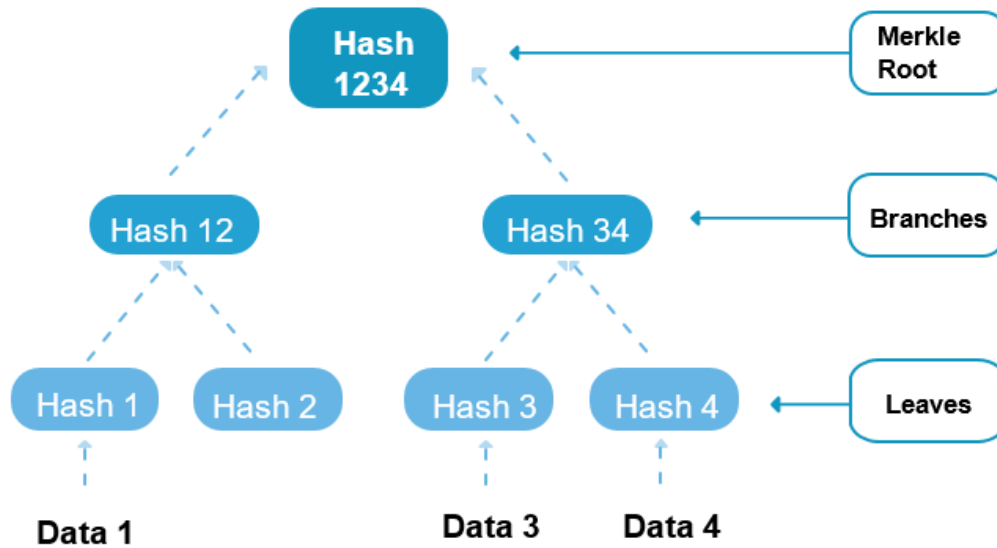


Figure 1.7: Hash Merkle Tree

and the Authentication Path Information “API” across branches layers. As example, “Hash1” could be authenticate by sending the leaf value “Data1” and the path “Hash 12, Hash 34”. Based on, the root node computes the hash1’ relative to the received value “data1”. Using the sent path, it calculates the hash value of the upper branch Hash12’ by hashing the pair (hash1’, hash2), and the root value Hash1234’ by hashing the pair (hash12’, hash34). Then, it compares the original merkle root with the calculated one. In case of equality, the data is verified.

1.3.6 Transaction

As Blockchain is a large database, the transaction is the operator that changes the state of this database. A transaction forms an independent unit of work that verify four main properties commonly referred as the acronym ACID [27]:

- “Atomicity propriety” means that a transaction is either fully performed or not at all.
- “Consistency” refers to satisfaction of database constraints after transaction.
- “Isolation” ensures that each transaction is executed in an independent way.
- “Duration” highlights the sustainability of the transaction effect once completed.

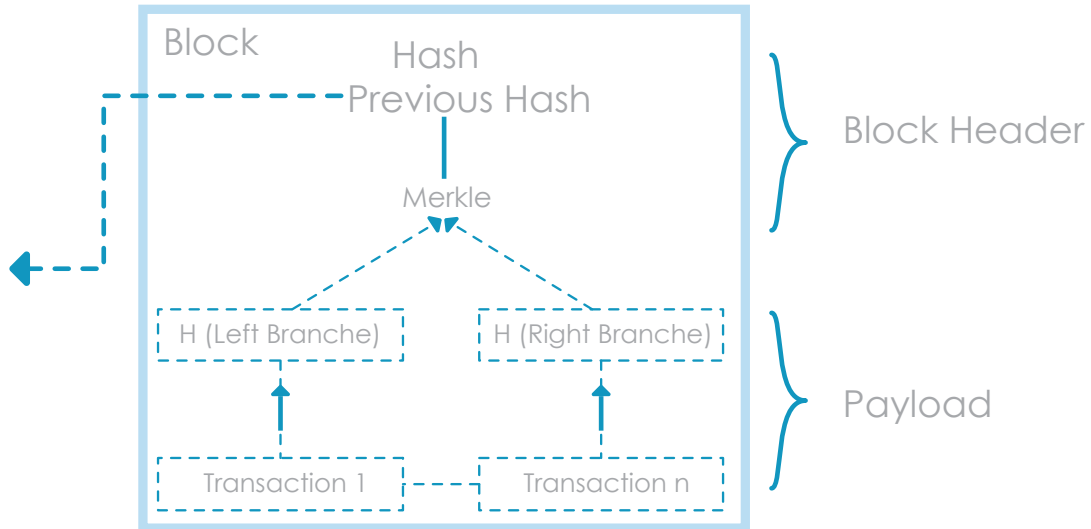


Figure 1.8: Block

1.3.7 Block

A block is defined as a unit encompassing a batch of transactions. Blocks can be chained similarly to elements of a hash chain and form a Block Chain (Blockchain) in this way. A chain of transactions leads to a very long chain; therefore, it is more efficient to make blocks of the units of the hash chain. As all transactions must be shared with all the actors interested in them, publishing one transaction at once is not efficient. Rather, a block of several transactions should be announced [28]. The block is divided into a block header and a payload. The block header includes the metadata of the block, namely, a timestamp, the Merkle root derived from the payload, and the hash pointer to the previous block [29]. The payload includes the actual transaction data. As illustrated in Figure 1.8, the payload is made under the Merkle tree and a block integrates units of a hash chain. Storing all the actual data will not increase the efficacy and maintain the integrity level of a chain. Therefore, the hash of a block represents only the hash of the block header [30]. The Merkle root changes meaning and the block hash changes as well when the transaction is tampered with, which is the case of the violation of the chain integrity.

Blockchain

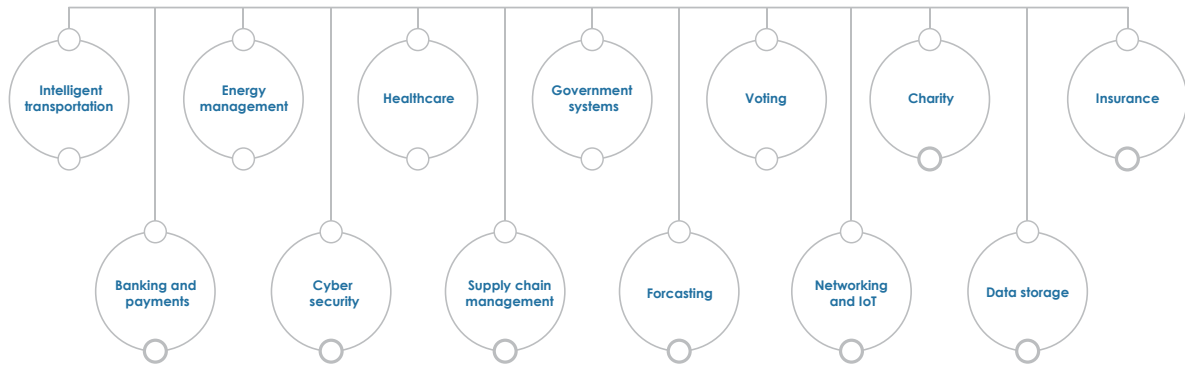


Figure 1.9: Blockchain Applications in Different Domains

1.3.8 Peer-to-Peer Network

A peer-to-peer network represents a collection of loosely coupled interacting autonomous nodes. Due to its decentralization, it is possible for nodes to join and leave the network unimpeded. A pure peer-to-peer network is the one in which all participating nodes have the same privileges [31]. Typically, the nodes share resources. A seed node in a Blockchain is the node that is known, which enables joining the network [27, 32].

1.4 Blockchain uses cases

The Blockchain technology has integrate many industries as illustrated in Figure 1.9 and is targeting many others in next decades. In the following, we introduce briefly some examples of these industries:

- The banking and payments sector gives access to financial services even including countries without - traditional banking. Payment and others financial operations become faster, more efficient and secure.
- The cybersecurity sector helps in verifying and securing data using advanced cryptography. The

1.4. BLOCKCHAIN USES CASES

data is less prone to be hacked or changed without authorization. There is no more need for an authorized third party or a middleman as in traditional legacy systems.

- Supply chain management sector helps in documenting transactions in a permanent decentralized record and monitoring them in a secure and transparent way. This allows decreasing delays and further human mistakes, it is also used to minder costs, labor, etc in the supply chain and to verify the authenticity and the fair trade status of products.
- Forecasting provides a decentralized market for consulting, analysis and forecasting operations invariant domains such as election, sport, stocks market and energy consumption.
- Networking and IoT market proposes decentralized networks of IoT devices using Blockchain. Operating like a public ledger for a large number of devices, the need for a central entity to handle all IoT communications devices is eliminated.
- The global insurance market is based on trust management. Since Blockchain presents a new way of managing trust, it can be used to verify many types of data in insurance contract like the insurance person identity.
- Online data storage using Blockchain allows cloud storage to be more secure and robust against attacks, hacking, data loss or human errors.
- Charity using Blockchain is more sure that financial aids and donations reach those who deserve it, and helps in fighting inefficiency and corruption.
- Voting presents the most society area where Blockchain contributes. It can be used for voter registration, identity verification and electronic vote counting to make sure that legitimate votes are counted and no voice was changed or moved. Immutable, publicly-viewable ledgers of recorded votes would make elections more fair and accurate.
- Government systems are often slow and prone to corruption. Blockchain can reduce bureaucracy and increase security, transparency and efficiency of governmental operations.
- Healthcare is another industry that relies on legacy systems. Hospitals need a secure platform to store and share sensitive data to avoid hacking and privacy breach problems. Blockchain helps

1.5. HISTORY OF BLOCKCHAIN

safely storing medical records and sharing them with authorized users. It helps improving data security, accuracy, and speed diagnostics.

- Energy management has used to be a highly centralized industry. Energy producers and consumers interact with each other through public grids or trusted third intermediary. With Blockchain, a decentralized system of buying and selling energy is established. It could be also adopted for products retail, real estate and similar commercial activities.
- The intelligent transportation industry has known interest revolution owing to Blockchain. It helps to implement a secure and trust ITS infrastructure based on peer-to-peer networks. Blockchain-based IoV solutions (BIOV) allow drivers and users to arrange transport conditions, share and update road and infrastructure events and status in a secure way without third party providers. Automatic parking payments, tolls, and fuel are also provided.

As well, researchers were interested in Blockchain technology application to this last cited industry. Contributions are related to traffic management, driving safety, road safety and security, payments and billing, parking services, privacy, and security preserving for ITS, Etc. After discussing the chronological Blockchain evolution, we will present a detailed overview of the existed related researches.

1.5 History of Blockchain

The history of Blockchain technology puts lights on its beginnings and evolution leading to apparition of smart contracts, Ethereum, and Hyperledger implementations. As illustrated in Figure 1.10, three basic periods are distinguished: the Pre-Bitcoin englobes cryptographic science and related areas. The Blockchain 1.0 determines first implementation of the famous financial application, Bitcoin. Finally, the Blockchain 2.0 announces more elaborated Blockchain platforms as Ethereum contracts and Hyperledgers.

1.5.1 Pre-Bitcoin

Some researches [12–14, 33, 34] associate the Blockchain technology with the apparition of Bitcoin, proposed by Nakamoto in 2008 [35]. However, it existed earlier than that. First works were elaborated on cryptographic secure chain of blocks in 1991 by Stuart Haber and W Scott Stornetta [36]. The

1.5. HISTORY OF BLOCKCHAIN

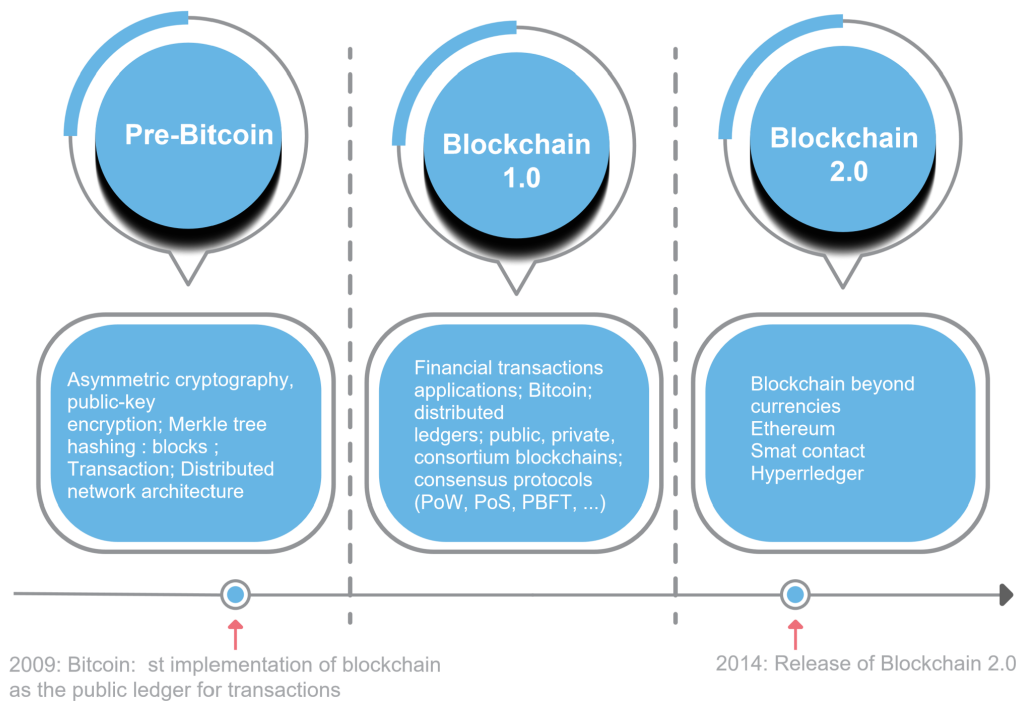


Figure 1.10: History of Blockchain technology

idea was to implement an anti-fraud system against tampering timestamps of data. In 1998, Szabo et al. [37] proposed a decentralized digital currency mechanism called “bit gold” that presented an introduction to what is called later “Bitcoin”. However, “bit gold” was never implemented. Two years after, Konst et al. [38] published a unified theory of encryption protection chains as well as some implementations. In 2008, Satoshi Nakamoto published his white paper “Bitcoin: A Peer-to-Peer Electronic Cash System” [35] in which he offers a new form of digital currency, the Bitcoin that was implemented in 2009. The evolution of this technology has been carried out, which lead to the born of more generic Blockchain 2.0 that refers to applications beyond cash transactions and currencies. In 2015, smart contracts and Ethereum Blockchain [26] are then evolved.

1.5.2 Blockchain 1.0

The Blockchain 1.0 phase refers to the launch of Bitcoin Blockchain application. In this section, we explore in close the Bitcoin standard, used consensus protocols, and rewarding miner’s mechanism.

1.5.2.1 Bitcoin

Bitcoin [12] is a public, decentralized and fully distributed peer-to-peer system for digital currencies. Originally designed by Satoshi Nakamoto alias, bitcoin aimed to create an electronic cash solution sheltered from any central authority for validation or settlement of transactions and currency insurance. Unlike traditional currencies, it is entirely virtual, where no physical or digital coins are handled. Bitcoin users possess keys that prove ownership rights in the network. Users sign transactions with their keys to unlock the value and spent it by transferring to another owner. Often, keys are registered in a digital wallet on users' terminals. Bitcoin relies on a robust computation process, called "minning" that verifies and validates unanimously a transaction during every ten minutes on average. Miners are then rewarded. Owing to bitcoin, the problem of double-spent transactions of digital currencies is no more addressed.

1.5.2.2 Consensus protocols

Consensus protocols present the core of Blockchain technology. Their relative main role is to maintain and verify transactions across a non-full trusted and distributed network using cryptographic mechanisms. Besides, consensus nodes are able to validate transferred data even by approval or declines. Validated data are also appended in an ordered way into the Blockchain register. In the following, we expose the most known studied consensus protocols in the literature.

1.5.2.2.1 Proof-of-Work (PoW) The Proof-of-Work (PoW) is among the most widely used consensus mechanism in existing Blockchain implementation [39]. Each modification of the single chain elements requires subsequent modification of all upcoming elements in order to ensure validity. Accordingly, dishonest nodes aiming to change previous blocks are forced to work harder than honest nodes, which strive to extend the current chain with most work behind [40]. Consequently, the security of the Blockchain is enhanced by this mechanism. Mathematical puzzles can be regarded as PoW, as the first miner who finds the solution is permitted to publish the block. Some puzzles are very heavy in a computational sense, as they require performing numerous computations to solve the puzzle. Therefore, miners with advanced computational capabilities have better chances. To summarize, the probability of solving the puzzle first corresponds to the miner's proportion of work and contribution. This property is labeled as progress-free. Thus, the number of blocks created by a minor

is proportionate to his contribution toward solving the puzzle [41]. The consensus mechanism is the extension of the branch with the most computations behind it. The longest branch is the branch with the most work behind when blocks are mined with the same interval and with the same puzzle difficulty. Consequently, the consensus mechanism leads to a long-term consensus chain.

1.5.2.2.2 Proof-of-Stake Proof-of-Stake (PoS) represents a consensus mechanism based on a proof of ownership which might be at stake [42, 43]. Intense computational work is not needed [44, 45]. By using PoS, the miners mine blocks proportionally to their stake [46]. There are a few algorithms involved. PoS is comparable to PoW as the miners mine blocks directly proportionate to their wealth, rather than using money. Similar chosen miners may mine with same probability proportionate to its stake. Thus, the similar miner is chosen to propose a block if the selected miner does not do it on time [47]. The consensus mechanism is based on the Blockchain with the most work behind it, which is the most stake.

1.5.2.2.3 Delegated Proof of Stake (DPoS) Extended from PoS, the DPoS [27, 31] is a dependable verifiable and transactions approval protocol based on shareholder voting scheme. The DPoS algorithm chooses, by distributed vote, contributor nodes that will play witnesses and delegates roles in the validation process. Elected witnesses are called to generate blocks regularly, every defined time slot. Delegates nodes are in charge of deciding or modifying Blockchain parameters such as intervals of blocks, sizes of transactions, transactions fees, transactions per block, Etc, to ensure reputation. Non-trusted nodes could be rejected progressively. DPoS saves more energy and accelerates transactions rate comparing to PoS [32].

1.5.2.2.4 Practical Byzantine Fault Tolerance (PBFT) Originally introduced in the late 90s, PBFT [31] is a powerful consensus algorithm that fix errors transmissions designed for consortium Blockchain where members are partially trusted. It has proved efficiency in asynchronous systems. In the following, we explain how PBFT consensus decision is made: first, a client node requests a timestamps for his “request message” from the master node. In his turn, the last records the “message request” and tag it with a reference number then broadcasts a “pre-prepare message” to other miners. In this stage, each miner is free to accept or deny the alert. In case of acceptance, all involved miners exchange a “prepare message” between each other’s until collecting $2f+1$ ones. Then, each involved

1.5. HISTORY OF BLOCKCHAIN

node sends a “commit message” to the master in order to inform readiness for executing the “request message”. After executing the “message request”, miners send the “reply message” to the client node. For some network delays, the result could not reach the destination node, in this case, the client resent his request again. Since the request has been treated, answers will be resent. This algorithm has a polynomial complexity.

1.5.2.2.5 Federated Byzantine Agreement (FBA) FBA [17, 48, 49] is a consensus algorithm useful for multi-joining of dishonest nodes scenarios aiming to over number trusted ones and provoking the problem of general Byzantine. FBA rely on quorums which are node sets grouping quorum members and quorum slice. A node is free to choose a trustworthy third party, referred as quorum slice, to validate a transaction. Nodes could be part of more than one quorum leading to intersected ones. Stellar [50] and Ripple [51] are FBA model variant protocols.

1.5.2.2.6 Proof of Authority(PoAu) PoAu [27] is an identity-stake based consensus mechanism. Nodes entrust authorized members as validators. One block is considered as accepted if most validators confirm it. New authorized members could join the network by voting. This consensus model tends to be centralized and is popular used for energy industries such as Energy Web Blockchain or in secured and full integrity environments.

1.5.2.2.7 Proof of Elapsed Time (PoET) Often used on permissioned networks, PoET [27] is a scalable, nimble cost and energy-efficient consensus protocol designed by Intel’s Sawtooth project. It is based on random blocks generation with preventing high utilization of processing resources or coins and avoids greedy consumption of energy as electricity for example. Since all participant nodes are equal to be a miner, PoET follows a fair mechanism for selection: all nodes are given a random waiting time during which they are on standby. The node that finish its waiting time first is then selected to generate the block. PoET has to ensure that waiting times assignment among nodes is purely random and should verify that the winner node had completed really its waiting period of time. PoET requires mutual trust to keep the environment protected [52].

1.5.2.2.8 Proof of Activity (PoAc) PoAc [27] is a consensus algorithm for decentralized systems. It is a hybrid protocol, combining blocks generation through the PoW mining mechanism and validation

process by the PoS mechanism.

1.5.2.2.9 Proof of Burn (PoB) PoB [27] is a consensus algorithm for state agreement and validation of Blockchain networks. It is considered as PoW alternative and aims to prevent double spending of cryptocurrency coins. To become a validator blocks node, coins must be paid to get that right in return. On the other hand, validated coins will be burned or destroyed. Because the validation process is costly, PoB avoids unnecessary money and resources waste.

1.5.2.2.10 Proof of Capacity (PoC) PoC [27] is an energy saving consensus protocol. In order to gain next block production, a concurrent validator node has to engage hard drive spaces to host outcome data named “plots”. Besides, it does not require expensive ASIC hardware. Proof of storage and proof of space are variants of this model.

1.5.2.3 Mining Incentives

The security of the Blockchain is based on the incentives for the miners to follow the protocol. There must be incentives offered to generate and validate blocks with appropriate transactions issued by the network and to work on the branch with the most work behind it. One example of an incentive is monetary, as the miners are rewarded for mining blocks leading to the long-term consensus chain [13]. On the other hand, they are penalized if not complying with this rule. The penalization in PoW involves the reduction of the power needed for the computations [47], while in PoS the penalization occurs through the stake [42,46]. Without penalization, miners might mine on different chains in the same time and collect rewards accordingly, which would increase the profitability of mining malicious blocks. This problem is known as Nothing-at-Stake, considering that the miner does not lose anything by mining on different chains [46,53]. Therefore, it is necessary to have a penalization for mining on the blocks which are not the part of the final Blockchain. Due to incentives and penalizations, miners’ profit from mining on the blocks is expected to belong to the true Blockchain. Considering that the majority is honest, it is consequently profitable for a node to be honest [13,44]. The rationale behind PoS is that the stakeholders find their stake important, and thus, receive incentives to secure the system [46]. Furthermore, monetary incentives ensure following of the prescriptions. There are also business and social incentives, such as a consortium with known participants running a Blockchain.

Incentives contribute to honest business and social relations, which ensure successful collaborations.

1.5.3 Blockchain 2.0

Blockchain 2.0 phase refers to the birth of Blockchain applications beyond commodities activities. We expose and discuss in this section further wide range Blockchain implementations developed after Bitcoin, like Ethereum, and Hyperledger.

1.5.3.1 Smart Contract

The smart contract represents an executable piece of code which might reside on a Blockchain, so that the script can be inspected by all participants. A smart contract is comparable to stored procedures in conventional relational databases. The difference is that the smart contract resides on the Blockchain [54]. Therefore, a stored procedure is not necessarily enforced, whereas it is not possible to bypass the smart contract [55]. It is executed on all the nodes; hence, each node runs a virtual machine. Accordingly, the Blockchain represents a distributed virtual machine. Considering that the code is executed on every node, it is necessary to avoid inconsistencies by having a precise, deterministic contract. Put shortly, the smart contract is an autonomous actor behaving transparently and predictably [14, 55].

1.5.3.2 Ethereum

Blockchain technology has been applied to various applications labeled as Bitcoin 2.0, Blockchain 2.0, and Crypto 2.0. Ethereum [56], established in 2015, is the biggest open-ended decentralized software platform (DApps). It allows building and running without fraud, downtime, interference or control by a third party. It also represents a programming language (Turing complete) based on a Blockchain. Developers use it for building and publishing distributed applications. The Ethereum applications are diverse and run on its platform-specific cryptographic token, Ether. Ethereum had launched a pre-sale of Ether in 2014, and it received considerable attention from developers interested in developing and running applications inside Ethereum. The use of Ether is two-fold. First, it is used for trade a digital currency exchange like other cryptocurrencies. Second, it is used within Ethereum to run applications and to monetize the work. As defined by Ethereum, it is employed to “codify, decentralize, secure and trade just about anything.” Ethereum’s big project is Microsoft’s partnership

with ConsenSys offering “Ethereum Blockchain as a Service (EBaaS) on Microsoft Azure.

1.5.3.3 Hyperledger

Hyperledger [18, 57] began in 2015 under the Linux Foundation. The idea was to create an open source Blockchain technology making individuals, businesses and interested parties work together. It is a modular, highly secure and interoperable distributed ledgers solutions involved in concrete domains such as banking, financial services and healthcare. The Hyperledger project comes up on a set of frameworks and tools. As framework examples, Hyperledger Burrow [23, 57] presents a strongly deterministic and permissionable smart contract machine which offers both access control and authorization layers to clients. Originally developed by Monax [58], it was classified as the fourth distributed ledger platform within Hyperledger in 2017. The second framework is called the Hyperledger Fabric [24, 57]. It is a modular, scalable and flexible platform for developing permissioned distributed ledgers solutions. Its ability to support varied consensus protocols makes it suitable for different trust models and uses cases. Contrarily to others platforms, Hyperledger Fabric does not need a specific coding language for a specific domain or cryptocurrency to run applications, it is then called a general-purpose programming languages platform. It allows forming participants channel for creating separate ledger where transactions are hidden for other participants in the same private network. This is useful in case of competitor participants. Furthermore, it allows portable membership for permissioned models. Moreover, Hyperledger Indy [25, 57] is a decentralized identity built distributed ledger. It allows digital and interoperable identities creation and use on distributed ledgers or Blockchains. Hyperledger Indy meets requirements of privacy and self-sovereignty of identity. Identity claims could be verified by combined or individual secured transferred information such as passports, birth certifications, driver licenses, etc. We introduce also the Hyperledger Iroha [15, 57] which is an easy incorporated-in-project distributed Blockchain framework. It was originally developed by Soramitsu and proposed by Soramitsu, Hitachi, NTT Data, and Colu to Hyperledger. It benefits of a simple structure, gives attention to mobile application development and uses new chain-based Byzantine Fault-Tolerant consensus algorithm. We present the Hyperledger Sawtooth [29, 57] which is a modular framework aiming to preserve the distributed structure of ledgers and safety of smart contracts. It allows organisations and user groups to evaluate their Blockchain applications. It enables dynamic consensus where consensus algorithms could change easily. It supports Proof of elapsed time (PoET) consensus protocol.

1.5. HISTORY OF BLOCKCHAIN

Parallel execution of transactions and their privacy are also reserved. As utility libraries and tools provided by Hyperledger, we name the Hyperledger Caliper [57, 59]. It is a performances evaluation tool for Blockchain implementation, compatible with many Blockchain platforms. It is used to measure transaction latency indicator, transactions per second (TPS) indicator, resource utilization and others metrics. Furthermore, Hyperledger Cello [57, 60] is another toolkit providing an on-demand deployment model of Blockchain system where a real-time dashboard is provided to users in order to check Blockchain system status and statistics (e.g. events, system performances and utilization) and manage Blockchain and chaincodes. It helps interested parties to create and manage easily and rapidly their Blockchain systems. Python and JavaScript are its main programming languages. It exists also the Hyperledger Composer [57, 61] which allows quick and simple creation and integration of Blockchain applications and smart contracts. It presents an open runtime tool-set, very useful for business networks where users are defined via their unique identities and could belong to others business networks. We also present the Hyperledger Explorer toolkit [57, 62] which is a viewing dashboard that enables network information control of transactions, blocks, logs, etc. This tool is compatible with open source, commercial, authorization or authentication platforms. Finally, the Hyperledger Quilt tool [28, 57] is a ledger interoperability solution toolkit based on Interledger Protocol (ILP) [30] implementation, written in Java. ILP is a routing payment protocol that transfers values across both distributed and non-distributed ledgers. Most of these tools support Hyperledger Fabric Blockchain infrastructure.

1.5.3.4 A comparison between Bitcoin, Ethereum and Hyperledger

Table 1.1: Features comparison of the well-known Blockchain technologies: Bitcoin, Ethereum, and Hyperledger

	Bitcoin	Ethereum	Hyperledger
Type	Public	Public/private	Private
Application	Crypto-currencies	General platform	General platform
Blockchain platform	No	Yes	Yes
Source	***	Open source Ethereum foundation	Open source Linux foundation
Consensus algorithm	PoW	PoW,PoS	PBFT, others
Language	C++, Golang	Solidity, LLL, Serpent	Java, Golang
Currency transaction rate	Lower	Higher	No
Data exchange rate	No	Low data volume	High data volume
Energy saving	No	No	Yes

Bitcoin, Ethereum and Hyperledger present the three emerging Blockchain technologies [63, 64]. Features like types of Blockchain (public/permissionless, consortium, or private/permissioness), adequate consensus algorithms (PoW, PoS, PBFT, etc), suitable applications (currencies, smart contracts, etc) differ from one Blockchain environment to another as shown in Table 1.1. Bitcoin was the first popular implementation. It is a public Blockchain that uses a stack-based language and a secure hash algorithm, SHA-256. It was used basically for cryptocurrencies ensuring its tamper-proof in a public distributed ledger. Bitcoin primarily acts as a store of value and a medium of the payment transaction. However, the transaction rate per second is limited. Besides, Bitcoin adopts the PoW algorithm for consensus operations that requires high computational performances. Ethereum allows developers and clients of the Enterprise to access a single click cloud-based Blockchain developer environment. Like Bitcoin, Ethereum is also powered by the principle of distributed ledgers and cryptography. But, it uses Turing complete language and PoW algorithm as a secure hash algorithm. The purpose of Ethereum is to enable peer-to-peer contracts and applications through its currency vehicle. It does not aim at being established as a payment alternative. Rather, its main aim is to facilitate and monetize developers who are building and running distributed applications (DApps). Ethereum allows more transaction rate in both private or public Blockchain. Besides the PoW, it supports as well the PoS

consensus algorithm to moderate required computational complexity and to enhance performances. For these reasons, Ethereum presents a better solution for limited computational capacities and a higher transaction rate environment. Hyperledger aims to provide more improved Blockchain environment. It is a Linux open source platform designed for business applications. The Hyperledger Fabric presents the commonly used Hyperledger framework implemented on a private ledger. It supports more sophisticated consensus algorithms like PBFT, PoW, PoS, etc. It can ensure high transaction volumes around 3500 per second which makes it suitable for high data applications. It supports simple mechanisms of access control. Contrary to Bitcoin and Ethereum, the Hyperledger Fabric is not recommended for cryptocurrencies like public transactions or incentive approaches since it is based on permissionless environment.

1.6 Conclusion

This chapter defined the Blockchain technology with a careful chronological study of its evolution from the Pre-Bitcoin phase, symbolized by the fundamental cryptographic systems, to the Blockchain 2.0 phase, typified by the use of Hyperledger and Ethereum implementations and smart contracts.

Chapter 2

State of the art: Study and discussion of existed related works

Contents

2.1	Introduction	47
2.2	Internet of Vehicles (IoV)	48
2.2.1	IoV definition	48
2.2.2	The IoV layered architecture models	49
2.3	State of the art literature conferences: classification methodology, statistics and discussion	51
2.3.1	Literature Reviews	55
2.3.2	Security	57
2.3.3	Transports applications	59
2.3.4	Energy	60
2.3.5	Data management	61
2.3.6	Communication and networks	61
2.3.7	Payments and optimization	62
2.4	Open challenges	63
2.5	Conclusion	64

2.1 Introduction

This chapter provides a comprehensive literature review of Blockchain-based IoV (BIOV) applications, dividing the exiting studies into six categories: Security, Transport applications, Energy, Communication and network, Data management, and Payments and optimization. Section 2.2 thoroughly analyzes the IoV concept and examines all its existing layered architectures before discussing

the seven-layered architecture's acceptance. Section 2.3 elaborates the methodology used to select research papers from the database, presents useful related statistics, classifies and discusses the reviewed papers according to their research directions and the implemented IoV layer belonging, and analyses the surveys. Finally, Section 2.4 discusses current open challenges and issues related to the application of the Blockchain.

2.2 Internet of Vehicles (IoV)

2.2.1 IoV definition

The Internet of Things (IoT) represents a network of physical devices embedded with network connectivity, actuators, sensors, software and electronics, such as home appliances, vehicles, and other items. Through the embedded computing systems, each device is identified and incorporated into the Internet infrastructure. The network infrastructure allows remote control of such devices [65]. Accordingly, computer-based systems have been increasingly integrated into the physical world, which leads to economic gains, accuracy, and efficiency, and decreased human intervention. Owing to actuators and sensors, this technology represents the general class of cyber-physical systems, including smart cities, intelligent transportation, smart homes, virtual power plants, and smart grids. The IoT is transforming conventional vehicular ad-hoc networks (VANETs) into the Internet of Vehicles (IoV) [66]. The IoV represents the real-time data interaction between vehicles and between vehicles and infrastructures through smart terminal devices, vehicle navigation systems, mobile communication technology and information platforms that allow information interaction, exchange of driving instructions, and control of the network system. This concept has enabled easier collection and sharing of information about vehicles and infrastructures. It also allows data collection, computing and exchange with Internet systems and other information platforms. Recently, this concept has taken firm ground in reality. It is estimated that in the near future, 25 billion things will be connected to the Internet, and vehicles will account for a significant number [66]. So far, Intelligent Transportation Systems (ITS) in Japan and Europe have already implemented some forms of IoV technology, whereas 55,000 licensed rickshaws in New Delhi have been equipped with GPS devices [67]. Owing to the prompt development of communication and computation technologies, this concept has attracted enormous research and commercial interest. Moreover, smart vehicles have been increasingly connected to the Internet, other

2.2. INTERNET OF VEHICLES (IOV)

vehicles nearby, and traffic management systems. In this way, vehicles are being incorporated into the Internet of Things (IoT)

2.2.2 The IoV layered architecture models

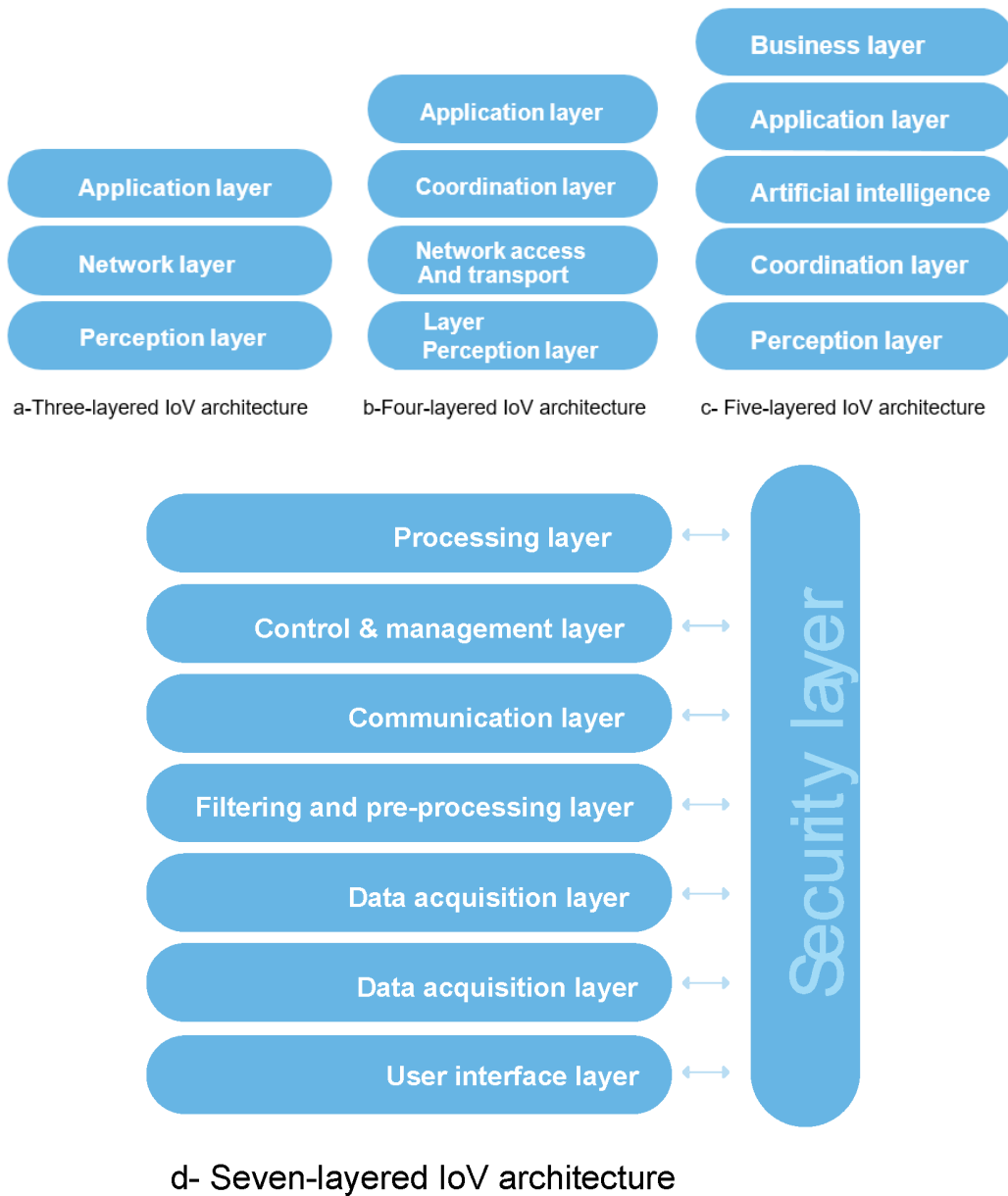


Figure 2.1: Current IoV layered architecture models

Precise definition of the IoV architecture model does not exist in the literature. Researchers

distinguish simple and more elaborated layered models, as shown in Figure 2.1. The three-layered IoV model introduces the basic levels of the architecture. The perception layer, named also sensing layer, represents the interface with the IoV environment that collects and communicates the events that takes place on the upper network layer. The network layer presents the connection point that transfers collected data through access networks like LTE, Wi-Fi and Bluetooth to the application layer. This last layer represents the decisional level that processes the received data using tools such as computational, statistical, analytical, and storage services. However, only three-leveled architecture has a broader perspective that does not describe specific functionalities of IoV precisely. The four-layered model adds extra functionalities to the network layer, such as control networks, data management (e.g., analysis and processing), and monitoring and node management. This model also introduces a coordination layer, responsible tasks such as for intelligent data processing and computing and resource allocation. However, this model does not specify the sets of functionalities for upper layers, particularly for data management. Hence, there is an urgent need to develop a more refined and more precise model. The five-layered model is global and clear and it includes the majority of IoV functionalities. Intermediate “network and transport” and “coordination” layers of the previous model are merged in one single layer in the five-leveled model, which is called the “coordination” layer. Its main role is the secure transportation of data through existing heterogeneous networks. Data management including storage, analysis, processing, and decision making is associated with the artificial intelligence layer. Efficient remote solutions for data management such as cloud computing are applied if the integrated computational resources of the IoV nodes are restricted and insufficient. The application layer introduces smart vehicular services to end-users and the business layer offers a practical display of results, like flowcharts, tables, and graphs, provided by the lower layer. This top layer contributes to determining the best business model or strategies. Although the five-layered IoV architecture model is regarded as the most structured, some gaps were identified, such as the lack of a security layer that ensures the smoothness and security of IoV functionalities of different levels, the lack of a communication layer that selects the most appropriate transmission channel if many heterogeneous networks are available (e.g., satellite, mobile network, and WiFi). To enforce the robustness of such a model, adding a preprocessing and a filtering layer allows lightening the load of the upper layer by preventing the transmission of redundant or unnecessary data. All these drawbacks were addressed by the seven-layered model. Subsequently, we describe each of its layers. The User Vehicle Interface

Layer is responsible for notifying the vehicle drivers about events occurring in the IoV environment by vibration, sound, or light signals. The Data Acquisition Layer collects data about these events. The Data Filtering and Preprocessing Layer analyses information and eliminates non-useful and redundant data. The Communication Layer uses qualified metrics to select the adequate heterogeneous network to transfer filtered data. As the name suggests, the Control and Management Layer applies control and management mechanisms such as data packet inspection, flow-based management, and policy enforcement. The processing layer computes the output of the received data according to predefined procedures and delivers the results to the end-users. Finally, the security layer reacts transversally with all previously mentioned layers. Its main role is the guarantee of security in all levels such as privacy, confidentiality, authentication, non-repudiation, integrity, and security against attacks. In the following section, we elaborate the seven-layered IoV architecture to specify research contributions.

2.3 State of the art literature conferences: classification methodology, statistics and discussion

Since this study focuses on ITS, in this section we present relevant statistics about research progression during the last five years. In the Scopus database [68], we found 275 research papers published from January 2015 to June 2020, when executing the following search request:

allintitle:

(blockchain|bitcoin|ethereum) (vehicle|cars|vehicles|transport|transportation|car|driver|Vehicular)

source: Scopus

At the moment of writing this document, the highest number of papers was published in 2019 (136 documents), as shown in Figure 2.2. The papers were published in well ranked journals and as conference proceedings, including IEEE Access, IEEE Internet of Things Journal, ACM International Conference Proceedings Series, and Computers and Electrical. This implies that Blockchain technology represents an attractive and hot topic for scientific communities. Figures 2.3 and 2.4 show that China dominates with excellence the research market. It occupies the first place with a total of published papers exceeding one hundred. The top Chinese faculties are the University of Electronics Science and Technology of China, Beijing University of Posts and Telecommunications, and Beijing Institute of Technology. The United State hold the second place with a total of published papers exceeding

2.3. STATE OF THE ART LITERATURE CONFERENCES: CLASSIFICATION METHODOLOGY, STATISTICS AND DISCUSSION

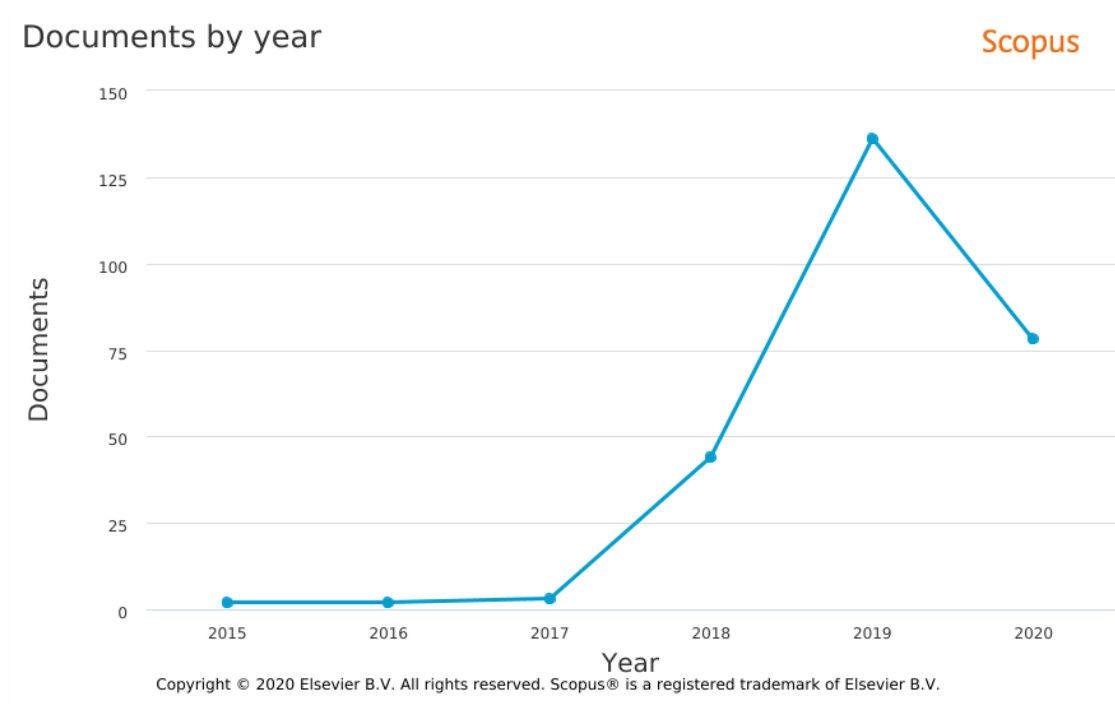


Figure 2.2: Number of published papers per Year

thirty. India is on the third place with 30 research papers, followed by South Korea, UK, Canada, France, Pakistan, Germany, and Australia, ordered respectively according to their contribution rate. According to the Scopus database, Y. Zhang from Texas A&M University (the US), N. Javaid from COMSATS University Islamabad (Pakistan), and S. Maharjan from University of Oslo (Norway) are most renowned authors in this field. To provide an overview of current research trends, Table 2.1 lists the first five most cited papers and does not include literature reviews. Publication year, publisher source, authors, and country are presented. The first paper [13] investigated the potential integration of Blockchain technology in transportation research. It introduced a seven-layered conceptual model helpful for standardization of typical Blockchain-based ITS architecture. The second paper [69] proposed Block-VN architecture for a distributed and secure vehicular network using Blockchain. This solution contributed to enhancing decentralized transport management. The third paper [44] focused on improving security for vehicular communications systems. More precisely, it proposed an efficient and secure key management framework applied to heterogeneous ITS and implemented on the top of a distributed Blockchain-based network. The fourth paper [40] proposed the CreditCoin solution that uses Blockchain to preserve privacy of users' identities within a distributed vehicular network

2.3. STATE OF THE ART LITERATURE CONFERENCES: CLASSIFICATION METHODOLOGY, STATISTICS AND DISCUSSION

and encourage users to participate in sharing traffic data via incentive mechanisms. Finally, the last paper [42] explored the security issues for electric vehicle cloud computing and edge computing to secure data and energy exchange for V2X communications. Specific cryptocurrency for vehicular applications was defined as data and energy coins. Blockchain-based consensus mechanisms were applied to achieve the proof of work of exchanged data and energy.

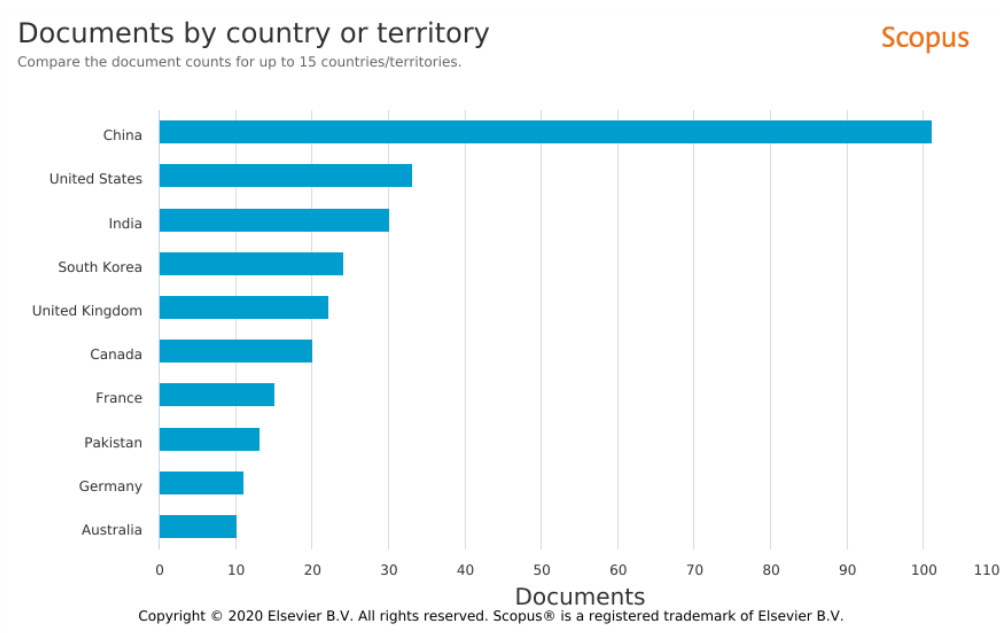


Figure 2.3: Number of papers by country

Our reviewing methodology was based on collecting all documents applying Blockchain in the IoV systems from the Scopus database [68] from January 2015 to June 2020. The selected papers were conformed to possess the following set of keywords “Blockchain, bitcoin, Ethereum, vehicle(s), car(s), transport, transportation, driver, vehicular”. A total of 275 papers was collected. Next, we filtered this primary set from unavailable or non-published documents. We also omitted economical papers, tutorials, chapters, and surveys.

The result set contains 211 documents that we classified into categories according to their research direction. We distinguished six categories, as presented in Figure 2.5:

36 % focused on the security axis, 18 % on transport applications axis, 16 % on energy axis, 13 % on communication and network axis, 13 % on data management axis, and 4 % of research works on payments and optimization axis. In the following section, we discuss research contributions by

2.3. STATE OF THE ART LITERATURE CONFERENCES: CLASSIFICATION METHODOLOGY, STATISTICS AND DISCUSSION

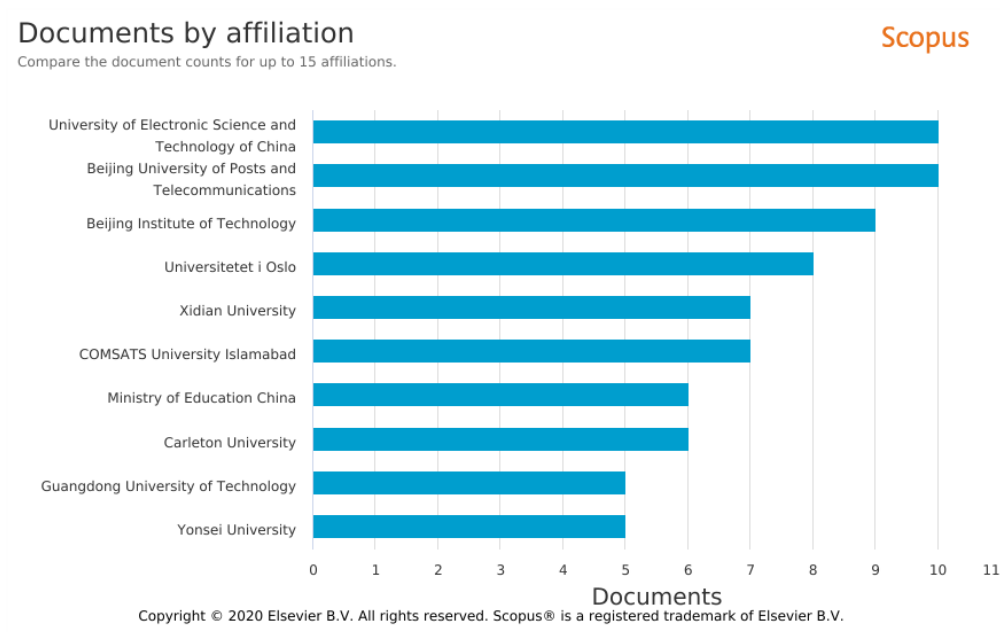


Figure 2.4: Documents by affiliation

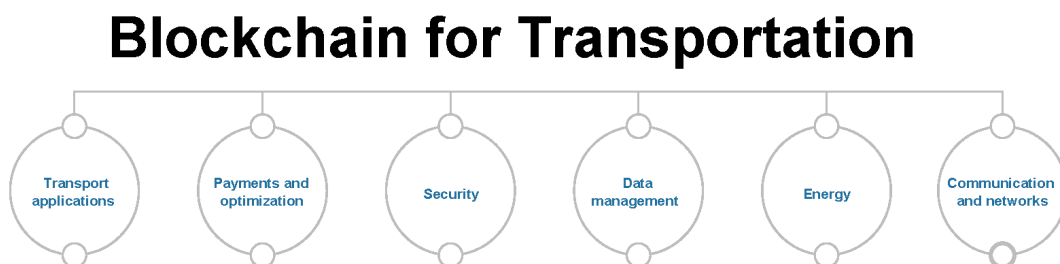


Figure 2.5: Blockchain for transportation

2.3. STATE OF THE ART LITERATURE CONFERENCES: CLASSIFICATION METHODOLOGY, STATISTICS AND DISCUSSION

category, projected on the seven-layered IoV architecture model presented previously and we present a summary of all reviewed works in Table 2.3.

Table 2.1: First five most cited research papers

Year	Title	Publisher source	Authors	Country
2016	Towards blockchain-based intelligent transportation systems	IEEE conference on intelligent Transportation Systems	Y. Yuan et al.	China
2017	Bloc-VN: A distributed blockchain based vehicular network architecture in smart city	Journal of Information Processing Systems	P.K. Sharma et al.	South Korea
2017	Blockchain-Based Dynamic Key Management for Heterogeneous Intelligent Transportation Systems	IEEE Internet of things Journal	A. Lei et al.	UK
2018	Credit Coin: A Privacy-Preserving Blockchain-Based Incentive Announcement Network for Communications of Smart Vehicles	IEEE Transactions on Intelligent Transportation Systems	L. Li et al.	China, KSA
	Blockchain-Enabled Security in Electric Vehicles Cloud and Edge Computing	IEEE Network	H. Liu et al.	China, Norway

2.3.1 Literature Reviews

Table 2.2 illustrate the relevant literature reviews on Blockchain technology published before July 2020. F. Casino et al. [70] investigated the state of the art of applications of Blockchain-based systems in different sectors, including healthcare, the IoT, supply chain, privacy, data management and business, and also clarified the use limitations, research gaps, and the perspectives of Blockchain. V. Astarita et al. [71] focused on Blockchain-based applications for transportation system, particularly the smart cities, the management of road traffic, and logistics and supply chains. This study specified current relevant research orientations, researcher gaps research, and commonly encountered challenges, and analyzed the maturity of the research progression. A. Ometov et al. [72] focused specifically on the application of Blockchain for smartphone devices. This study investigated the com-

2.3. STATE OF THE ART LITERATURE CONFERENCES: CLASSIFICATION METHODOLOGY, STATISTICS AND DISCUSSION

Table 2.2: The exhaustive list of current literature surveys on Blockchain technology

References	Title	Journal\Conference	Date of publication
[70]	A systematic literature review of Blockchain-based applications: Current status, classification and open issues	Telematics and Informatics	March 2019
[71]	A Review of Blockchain-Based Systems in Transportation	Information	December 2019
[72]	An Overview on Blockchain for Smartphones: State-of-the-Art, Consensus, Implementation, Challenges and Future Trends	IEEE Access	June 2020
[73]	A Survey on Blockchain Incentive Mechanism	ICPCSEE	September 2019
[74]	Blockchain applications in supply chains, transport and logistics: a systematic review of the literature	International Journal of Production Research	August 2019
[75]	Blockchain for the Internet of Vehicles towards Intelligent Transportation Systems: A survey	IEEE Internet of Things Journal	July 2020
[76]	Blockchain Technology for Cloud Storage: A Systematic Literature Review	ACM Computing Surveys	August 2020
[77]	Blockchain Evaluation Approaches: State-of-the-Art and Future Perspective	Sensors	June 2020
[78]	Blockchain for Industry 4.0: A Comprehensive Review	IEEE Access	April 2020

promise of worldwide but constrained energy devices in supporting Blockchain computing needs. It also discussed the relevant existing literature, the way of its implementation, the encountered challenges, and future views. A novel consensus protocol was also proposed and tested combining existing PoW, PoA, and PoS protocols. J. Huang et al. [73] focused particularly on the incentive functions of Blockchain technology, such as storage, computation, and transmission and addressed the state of the art current studies, challenges, and future directions. M. Pournader et al. [74] conducted research on Blockchain-based supply chain, logistics, and transport management. The authors classified the literature on the basis of four key domains: the technology direction, the trust direction, the trade direction, and the traceability or the transparency direction. They also discussed and concluded future research perspectives for each domain. M. B. Mollah et al. [75] investigated recent innovations related to BIoV systems. They examined the latest applications, main challenges, and research perspectives in this direction. P. Sharma et al. [76] reviewed the role of Blockchain, particularly the guaranteed trust feature, in securing cloud storage systems. S. Smetanin et al. [77] used strategies and key metrics to evaluate the Blockchain technology and defined its usage limits. Subsequently, they discussed related literature and highlighted major challenges and potential future works. They distinguished two main approaches to classify the reviewed literature: the analytical and simulation-based strategy and the emulations-based strategy. Evaluation metrics were classified into three categories: the Blockchain metrics are composed of consensus, the throughput of the transaction, and block size. The network size presented by its active nodes and the packet loss ratio are essential network metrics. More precisely, the node metrics regrouped hardware characteristics of the nodes, such as CPU/ GPU, memory, storage capacity, and read latency. U. Bodkhe et al. [78] reviewed the application of Blockchain in industry 4.0, discussed the work progression in this field, and referenced the used architecture with a focus on security solutions.

2.3.2 Security

Security issues are critical in vehicular networks because of their significant effects on the user's life. Security failures and hackers' threats can lead to cyber-attacks, resulting in vehicle immobilization, road accidents, financial losses, disclosure of sensitive data, and even endangering road users' safety. Numerous contributions were realized to enhance data security in this field, establish and improve one or more features of privacy, anonymity, authentication, trust, resilience to attacks, reputation,

immutability, confidentiality, integrity, accessibility, identification, transparency, and credibility. A distributed Blockchain-based trustful and credible platform was designed in [79] for the vehicular system. CreditCoin, privacy preserving announcement architecture was investigated in [40], to allow and encourage broadcasting messages while preserving identities. Users received incentive to sign and forward messages owing to the application of both the anonymous message aggregation protocol and a Blockchain-based incentive mechanism. A new credible data system was proposed in [80]. Data credibility was demonstrated from the reputation value granted to the sender. More precisely, the used Blockchain approach allows vehicles to evaluate the transmitted data based on their observations of the system. Subsequently, a score value is affected to the shared data and stored in a block. Owing to that score, vehicles could verify the reputation of the sender vehicles and the credibility of the sent data. A smart car authentication and revocation framework was designed in [81]. The objective was to use Blockchain techniques to improve updates of revoked vehicles' status and to decrease the processing and communications costs comparing to trusted authority central architecture. A study on the role of Blockchain technology in ensuring and enforcing authentication of both transmitted data and identities of vehicles was conducted in [82]. Another prototype presented in [83] explored how Blockchain technology could ensure transparency in the IoT-based distributed system. A trusted vehicle platform was proposed in [84] that allows distinguishing bad-behaved vehicles or malicious ones. Using Blockchain, the way of collecting the VANET environmental data, validation of packets, and participation of vehicles in generating and sharing blocks allow vehicles to gain or dismiss the trust of others. This platform also performed well in detecting Sybil attacks. A secure and transparent framework for CAV (Connected Autonomous Vehicles) was proposed in [85]. The proposed framework used a Blockchain technique to extract and store information reducing the risk of sharing fake users' requests, the alteration of information, or the compromise of smart sensors of connected vehicles. In [86], a Blockchain-based authentication mechanism using asymmetric keys and the Message Authentication Code (MAC) was proposed to improve privacy and authentication of messages for the VANET network. Practical Byzantine Fault Tolerance (PBFT) and Proof of Work (PoW) were used to aggregate consensus of authentication of messages. A combination of Blockchain technology and distributed cloud architecture was investigated in [87] to ensure privacy of vehicular data and secure them from attacks. This combined platform allowed massive and scalable data management, performing computational system, and offloading the computational and traffic load to the cloud, in

2.3. STATE OF THE ART LITERATURE CONFERENCES: CLASSIFICATION METHODOLOGY, STATISTICS AND DISCUSSION

Table 2.3: Research classification projected on research directions and IoV layer correspondence.

Research axis	Research projected to IoV layered architecture model		
	IoV Processing layer	IoV Communication layer	IoV security layer
Security	-	-	[40, 42, 79–83, 96–126]
Transport applications	[18, 88–92, 127–142]	-	[93, 143–146]
Energy	[54, 147–161]	[162]	[54, 163–167]
Data management	[44, 168–194]	-	[44, 168–194],
Communication and network	-	[69, 195–202]	[69, 195–202]
Payments and optimization	[41, 46, 47, 203]	[43], [45]	[41, 43, 45–47, 53, 203]

addition to security features provided by Blockchain, such as privacy, trust, and transparency.

2.3.3 Transports applications

Smart transports applications for IoV and vehicular systems are a wide, renewed and innovative market. Engineers have proposed and tested APIs as for congestion avoidance, traffic safety, in-vehicle entertainment APIs, mobility services as to locate, unlock, and read the odometer from cars across brands for example, etc. Researches related to transports applications for vehicles systems present varied contributions and are mainly localized in the IoV processing layer combined with the IoV security layer for security and privacy purposes. Blockchain based platforms for smart cars parking services, cars leasing, training and learning autonomous cars, secure selling and buying used cars, transparent dissemination of usage history of motors for trading purposes, were proposed respectively in [18, 88–91]. A smart contract based platform for emergent transport service was proposed in [92] where privileges of drivers and vehicles were saved, shared then deleted after the completeness of transport service. In order to protect data stocked in the in-built restricted resources of smart vehicles, a secure content caching scheme using private Blockchain and deep reinforcement learning approach was designed in [93]. A reliable framework for accurate GPS positioning system was designed in [94] on the top of a Blockchain architecture composed of in-built sensors vehicles, classic vehicles and roadside units. A credible traffic management mechanism that applies a group signature algorithm and ElGamal encryption algorithm to disable the transmission of malicious and fake messages between vehicles in a consortium environment was developed in [95].

2.3.4 Energy

With the introduction of smart cars, fully autonomous driverless cars, and EVs, numerous efforts were directed to energy and electric utility providers to digitally monitor, manage, and control their customers' EVs. Needs to retrieve the state of charge and the remaining range from an EVs battery, to schedule and remotely control the charging and discharging process and to optimize the relative pricing costs, and to build EV management dashboards were an urgent task for industries as well as researchers. The common application for current energy studies in the IoV is offering smart charging or fueling services in vehicular networks using decentralized, private, or consortium Blockchain [151, 152, 155, 157, 159, 204]. As examples of applications, a simple selection mechanism of the priceless charging unit is developed for EVs' drivers based on smart contracts [150]. A scheduling charging approach for EVs was proposed in [161] considering constraints such as the battery capacities, the rate of charging or discharging operations, and the relative cost of charging. This mechanism could be expanded to consider other selection criteria, such as actual battery status, traffic congestion, and service delay. Other works focused on designing efficient trading energy frameworks for EVs [54, 147–149, 153, 154, 156, 158, 160]. In [149], an optimized cost-aware trading energy platform was explored using a consortium Blockchain. This platform applied contract-based incentive mechanism to respect preferences of each EV and improve their participation cycle. At the level of the IoV communication layer, a novel distributed architecture for energy trading between specific bidirectional battery vehicles was proposed in [162]. An autonomous energy exchange process was achieved where surplus energy vehicles execute a discharging operation in order to charge low battery vehicles. At the level of the IoV security layer, studies were performed to secure energy transactions and protect the exchange mechanisms from attacks and security vulnerabilities [54, 163–167]. For example, In [163], energy exchange was secured using consensus algorithms over a topology composed of EVs and charging units. Smart contracts were also applied to personalize preferences and exigency required for charging services. In [166], the research contribution was based on a PoR consensus scheme to secure the delivery of energy in a private Blockchain-based energy vehicular network. Furthermore, an incentive scheme for a price model was also applied to order the charging and discharging process between energy nodes and energy-restricted nodes and to enhance the utility function of the energy exchange process and improve the satisfaction experience of users.

2.3.5 Data management

Smart cars and EVs with embedded computers, GPS receivers, short-range wireless network interfaces, and potential access to in-car sensors and the Internet are required to share and store collected events and sensitive data, such as identities of drivers or vehicles, cryptographic keys, localization, predicted future direction, and traffic and roads congestion. It has to be performed securely via a fully or semi-distributed vehicular network topologies to avoid security vulnerabilities and bottlenecks problems of centralized architectures. Concerning works focused on Blockchain based data management axis [44, 168–194], all contributions are based on both the IoV processing layer and IoV security layer to provide novel traffic services and/or to secure the transferred data and avoid security attacks among vehicular systems. Basic works propose smart frameworks for generating, storing, and sharing data over existing elements of the networks, such as vehicles, users, infrastructure nodes, and sensors, using a consortium or private Blockchains [169, 173, 180, 186, 187, 190, 191]. As a basic service to improve road safety, particularly with the introduction of self-cars-drivers, a control application of accidents was proposed in [169] that contributes to analyzing and verifying causes and guiltiness of occurred accidents. A privacy scheme was then implemented to prove the correctness of collected and stored events and registration of relevant drivers' information. In [180], a distributed solution for intelligent vehicular transactions was modeled and investigated to overcome the limits of traditional data management solutions based on the centralized approach. In [190], an enhanced version of the Diffie-Hellman algorithm was proposed to improve trust of the applied verification mechanism in a consortium Blockchain-based network. Besides, the used consensus mechanism was optimized to reduce consensus delays.

2.3.6 Communication and networks

The ability of vehicles to securely communicate with each other is key factor of successful vehicular systems. Studies communication and networks axis [69, 195–202], had the goal of aimed at enhancing or suggesting new communication protocols in vehicular networks such as VANETs [205, 206], IoT based networks [207], IoV [208–211], UAVs [212], SDN [99, 211]135], for better security, privacy, trust, reliability, authentication, anonymity, access control and security against attacks features. In general, the contributions focused on two major IoV layers: the communication layer and the security layer. M. Singh et S. Kim [213] proposed a smart and trustful communication protocol using Blockchain technol-

ogy for the cloud vehicular system. Successful exchanges were rewarded owing to the application of an incentive mechanism that increases the trust of the implied nodes. A hierarchical Blockchain system was implemented in [208] and it considers potential constraints of the IoV environment, such as a long delay to decide a consensus operation and the high topology dynamics of mobile vehicles. Besides, two ledgers were managed by two levels of the architecture to store circulating information according to its importance. A distributed network of Unmanned Aerial Vehicles (UAVs) was investigated to design a communication mechanism for generating and consulting transactions between UAVs. In [198], a hybrid 5G and cloud vehicle network was explored to support an emerging communication protocol for warning messages while ensuring privacy of sensitive users' information. In [201], a hybrid architecture was developed to secure vehicle-to-vehicle and vehicle-to- infrastructure communications. In [202], a hashing-based storage and access control scheme to manage data about traffic was proposed on top of a Blockchain-based vehicular network.

2.3.7 Payments and optimization

The payments and optimization category involves studies on untraditional, smart, and decentralized payment and billing solutions for IoV users that enable secure and efficient transactions, optimized price, and energy consumption. As for the IoV processing layer, the concept of a Blockchain-based billing service that secures transactions exchanged between EVs and charging stations was introduced in [46]. Another novel transaction structure, verification, and unique ledger registration mechanisms were proposed on top of an Hyperledger system to ensure trustworthily and tamper-proof of payments [41]. An original optimization model of the distributed scheduling mechanism of EVs' battery swap stations was investigate in [47]. The objective was to optimize the load and the cost of the generation of power. A smart contract-based rental car platform with optimized cost was also proposed in [203]. Concerning the IoV communication layer, a new topology composed of EVs and charging stations was designed to personalize Bitcoin transactions in a private network and reduce their correspondent costs and verification delays [43]. Another optimization approach [45] was conducted to maximize the throughput of transactions in a Blockchain-based IoV network under security and delay constraints. This approach uses Deep Reinforcement Learning to decide the adequate sizes, intervals, and producers of blocks that satisfy imposed constraints. Concerning the IoV security layer, all previously cited contributions of different IoV layers [41, 46, 47, 203] used Blockchain technology

to secure payment IoV services and ensure privacy, reliability, and/or authentication of transactions and shared data, such as payment records, users' identities, behaviors, and other sensitive information using Blockchain technology. For example, some consensus approaches such as the PBFT algorithm and smart contracts could be applied to verify transaction information.

2.4 Open challenges

In this section, we discuss some currently open challenges and issues relevant to the application of Blockchain. The first open issue is related to improving Blockchain solution performance regarding the time consuming limitation due to the time-consuming nature of the mining process. More precisely, the data must be mined in order to be shared in the Blockchain network. The duration of the mining process can reach up to 10 min [214]. Furthermore, the majority of the existing published studies employ Bitcoin as an implementation platform for Blockchain. Most importantly, as Bitcoin is essentially a cryptocurrency for selling and buying goods in a secure and anonymous way, therefore, smart contracts are not supported. Moreover, it does not possess the programming characteristics required for resolving computational problems aimed at allowing the transfer of various sensitive data. It is necessary to develop a Blockchain platform that is able to support Turing-Complete operations to consequently propose cutting-edge vehicle communication applications, which is the case with Ethereum.

The second actual open challenge investigates the combination of machine learning and Blockchain. Massive data are the key input for machine learning. However, there are some difficulties related to collecting the necessary amount of data. Since Blockchain allows the management and the storage of a huge volume of information in the forms of blocks, it could be useful to train machine learning algorithms. Primary works focused on the integration of Blockchain and machine learning aimed at decisions making improvement, result optimization, and prediction [45, 93, 99, 135].

The third open challenge explores the resource provisioning approaches and incentive solutions. The operation of the Blockchain is fundamentally based on the use of consensus algorithms, which require high computation resources to complete the block processing and validation on time and prevent delays. However, IoV entities suffer from limited equipped resources either for storage, networking, or computing. To overcome this resource constraint, studies explored dynamic mechanisms to optimize

the selection of mining nodes, prevent their saturation, or deploy other dedicated nodes for validation. Other approaches emphasize the deployment of incentive mechanisms to encourage candidates' nodes to lease their resources and participate in computing operations. The Blockchain system also uses important traffic signalization to approve transactions. Such network congestion may mismanage the existing resources and degrade the network performance. Hence, specific approaches to resources and traffic management are needed. Concerning the security issues, the key features that evaluate the security level of IoV networks were defined with metrics of privacy-preserving, reputation, trust management, security against attacks, and integrity. Blockchain has successfully contributed to reinforcing each of these features; however, other security issues associated with this technology need to be carefully considered. If hackers ever succeed to impose their control on the majority of nodes, the existing consensus algorithms becomes useless and inefficient, and blocks become prone to easy attacks. Accordingly, consensus protocols should be carefully tested before use. Another point of security weakness is figured in smart contracts, mainly presented as a simple code easy to attack. Finally, considering the scalability issues, critical performance investigations were mentioned due to the distributed and dynamic nature of IoV architecture. As the number of entities changes over time due to the insertion of pedestrians, drivers, autonomous vehicles, or infrastructure entities, new entities must earn trust of other participants. Consensus algorithms may contribute to this task to reduce ambiguity and foreignness between nodes, then, stabilize the whole network efficiency and functionality while scaling up or down. Moreover, massive and simultaneous events, such as attending a football match or evacuating damaged areas due to a disaster, could affect the current traffic management. IoV users become prone to delayed or uncompleted services, which harms the trust and reputation of the network, and hence, the quality of experience of users. Moreover, sudden events may generate unexpected traffic peaks and high transactions rates. This increase in data exchange should be well managed, particularly with the restricted resources of storage and processing of IoV entities.

2.5 Conclusion

After discussing different Blockchain applications in various domains, we focused on the intelligent transport applications for the IoV networks and classified the relevant studies into six directions, namely Security, Transport applications, Energy, Communication and network, Data management, and Payments and optimization. Next, for each direction, we classified the contributions according

2.5. CONCLUSION

to its belonging to the IoV layers. We noticed that most research contributions belong to three main IoV layers, individually or in combination, namely the Processing layer, Communication layer, and Security layer.

Chapter 3

Proposed solutions

Contents

3.1 Introduction	66
3.2 Blockchain IoT Solution for Vehicles communication (DISV)	67
3.2.1 System Overview	67
3.2.2 The Perception Layer	69
3.2.3 The Network Layer	73
3.2.4 The Application Layer	73
3.2.5 Nominal Scenario	79
3.2.6 Formal Testing Framework	80
3.3 Blockchain IoV Solution for payment (PSEV)	82
3.3.1 The Perception Layer	84
3.3.2 The Network Layer	87
3.3.3 The Application Layer	88
3.3.4 Nominal Scenarios	90
3.3.5 Formal Testing Framework	93
3.4 Conclusion	100

3.1 Introduction

In the last chapter, we gave attention to Blockchain-based applications in IoV. We have noticed that most existing solution has limitations regarding supporting the real-time operation and Turing-Completeness aspect. To address this challenge, this work introduces a Blockchain IoV Solutions with Real-Time Application (RTA) aspect aimed at establishing secure communications and payments between all participants in transportation systems.

3.2. BLOCKCHAIN IOT SOLUTION FOR VEHICLES COMMUNICATION (DISV)

This chapter comprehensively presents developed BIoV solutions. We explain the architecture and different software and hardware used components for both solutions and elaborate nominal scenarios. Furthermore, we introduce a model-based framework to validate the proposed approach. This framework is primarily based on the use of the Attack Trees (AT) and timed automaton (TA) formalisms used for testing the functional, load and security aspects.

The first section introduces the IoT Solution for Vehicles communication (DISV) aimed at facilitating the data exchange and cooperation between cars, infrastructure, and other actors of intelligent transportation systems. DISV allows the Internet of Vehicle networks users to receive messages and broadcasts them to the chosen Blockchain layer. In addition, the server confirms the received block based on its local knowledge and decides if it should be added to the smart contract.

The second section introduce “PSEV”, which is a Blockchain framework for vehicle communication and parking payment. PSEV consists of two modules, namely PSEV-Payment and PSEV-Communication. PSEV-Payment is a Blockchain-based solution that uses Ethereum to generate a smart payment system for parking. Furthermore, Blockchain IoV Solution for Vehicles Communication (PSEV-Communication) operates in real-time to ensure safe communication between all participants in transportation systems.

3.2 Blockchain IoT Solution for Vehicles communication (DISV)

3.2.1 System Overview

The purpose of this work is to present an IoV solution with Real-Time Application (RTA). This solution provides secure communication between vehicles and other actors in transportation systems. It attempts to overcome limitations such as execution time and accordingly, improves performance. A prototype of DISV was developed and tested it based on the following scenario: if a driver is drowsy, the nearest cars should be alerted by sending a message via Blockchain. Since it is based on an IoT architecture, the proposed solution should contain mainly three layers; the perception, the network, and the application layers, as illustrated in Table 3.1 and described below:

- The perception layer is the physical layer. It consists of several IoT devices equipped with sensors designed to identify and collect informa-

3.2. BLOCKCHAIN IOT SOLUTION FOR VEHICLES COMMUNICATION (DISV)

tion about the environment (i.e., physical parameters) and to detect nearby smart objects. The Android Application for Vehicles (AV) embedded into the perception layer collects and analyze data about the trip, the vehicle, and the driver's behavior. Android Application for Infrastructure (AI) simulates the role of IoT devices integrated into the roads such as radars, traffic lights, roadside electronic signs and others.

- The network layer connects the sensors to other servers, network devices, and smart things, and also transmits and processes sensor data.

- The application layer consists of Blockchain application and Central Cloud Server. It delivers application-specific services to the IoT devices. More precisely, the Blockchain application manages communication between vehicles and other actors in the transportation system. The Central Cloud Server is in charge of processing and analyzing the obtained data and managing invitations of other actors.

The Figure 3.1 depicts the architecture of the proposed solution and demonstrates the principal workflow, which contains three main steps. First, the cars send data to the central server (1). Second, based on the received data, the central server sends an invitation to connect to the Blockchain layer (2). Third, the cars can share the data with the others participants of the IoV in the same area securely (3).

3.2. BLOCKCHAIN IOT SOLUTION FOR VEHICLES COMMUNICATION (DISV)

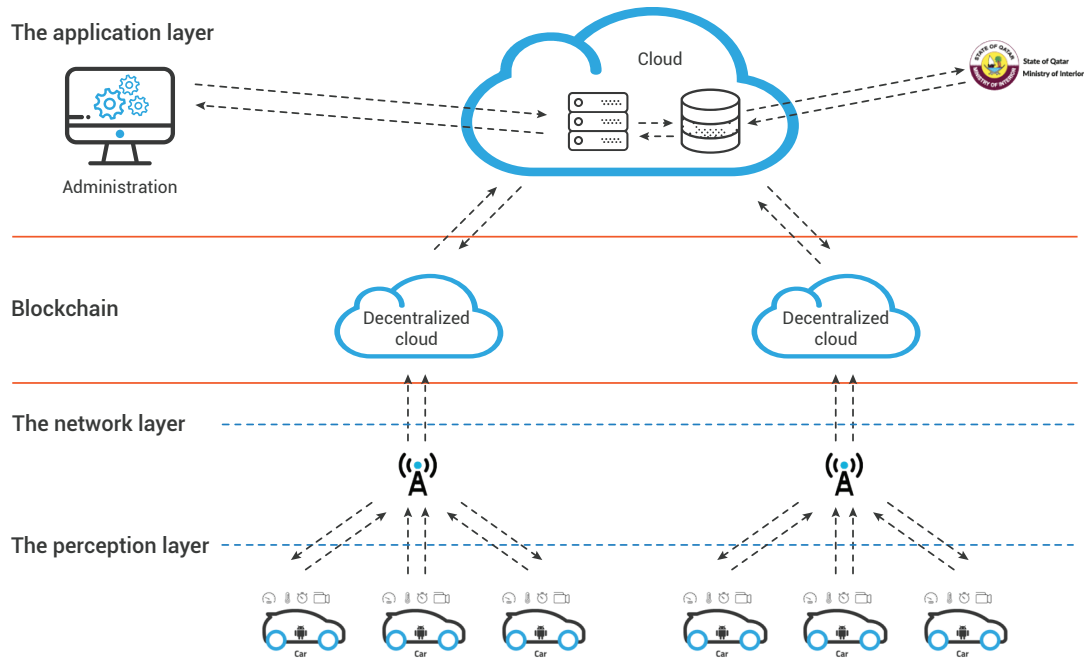


Figure 3.1: The architecture of the proposed Internet of Things solution.

Table 3.1: The main features of the developed IoT solution.

Layers	Developed Solution	Main Features
Perception layer	Android Application for Vehicles (AV)	Collects and analyze data about the trip, the vehicle, and the driver's behavior.
	Android Application for Infrastructure (AP)	Simulate the role of IoT devices integrated into the roads such as radars, traffic lights, roadside electronic signs and other.
Network layer		Connects the sensors to other servers, networks devices and smart things.
Application layer	Blockchain Application	Managing communication between vehicles and other actors in the transportation system.
	Central Cloud Server	Processes and analysis obtained data Manages invitations of other actors.

3.2.2 The Perception Layer

In order to test possible scenarios involving various components, an Android applications has been developed in the Android Application for Vehicles (AV) and for infrastructure (AP) as detailed in the

3.2. BLOCKCHAIN IOT SOLUTION FOR VEHICLES COMMUNICATION (DISV)

method to increase the number of training points is to use data augmentation. Codebox [215] was used to generate new images by performing a set of augmentation operations on the images extracted from video frames.

- Step 3 – Extracting facial landmark coordinates from images: In order to locate and represent salient parts of the face such as eyes, mouth, nose and the jawline, Facial Landmark is used. These landmarks are essential for head pose estimation, blink detection, yawning detection, etc. An open-source C++ library called Dlib [216] has pre written functions that are used to obtain facial landmarks. This library is programmed to find the x,y coordinates of 68 facial landmarks to map the facial structure as shown in Figure 3.3.

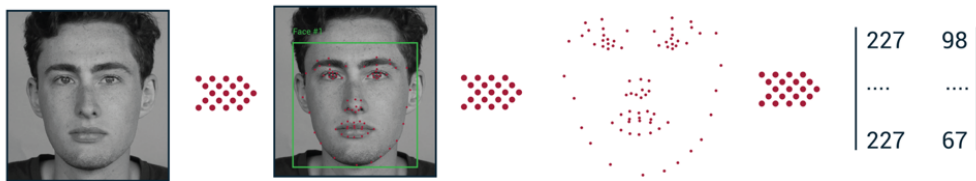


Figure 3.3: Facial feature detection

- Step 4– Training the algorithm: The landmark coordination extracted from images will act as the input to the algorithm, detailed in Algorithm 1, based on Convolutional Neural Networks Classifier with three hidden layers. During this step, a process training will ensue where there will be various predictions from which a model will be formed; corrections are made to the model if the predictions go wrong. The training will be processed till the wanted level of accuracy is reached.
- Step 5– Model extraction: Finally, the algorithm can decide if the driver is drowsy or not based on his or her face landmark. The trained model is saved as a file; so it can be used in the mobile application. The experiments were conducted on a Dell workstation with Intel Xeon Gold 6128, 3.4 GHz, 64 GB RAM, and NVIDIA QUADRO P4000. The accuracy results per driving scenarios using the new technique D2CNN-FLD method are shown in Table 3.2 ; its overall accuracy rate was 83.3%.

3.2. BLOCKCHAIN IOT SOLUTION FOR VEHICLES COMMUNICATION (DISV)

Algorithm 1: Real-Time Driver Drowsiness Detection Algorithm with CNN

Input: DLib Facial landmark positions and labels

Output: 2 - Class probabilities of drowsy driving

0: Loading the Data

0: Min-Max Scaling

0: Adding hidden layers and dropout:

a. The first convolutional layer with linear activation. { //The input layer consists of $67*2=134$ nodes with a total of 100 neurons.}

b. A LeakyReLU function { //The alpha rate is set to 10%.}

c. A MaxPooling function

d. Dropout is set to 25.4% to prevent over-fitting.

e. For $j=1$ to 3 do

(i). Adding convolution hidden layer with a linear function. { //The number of neurons used is 1024.}

(ii). A LeakyReLU function. { //The alpha rate is set to 10%.}

(iii). A MaxPooling function

(iv). Dropout is set to 20% to prevent over-fitting.

f. Final fully connected layer with a softmax function to get a 2 class output probabilities.

0: The model is trained until acceptable accuracy. =0

Table 3.2: Accuracy per driving using CNN method

Category	Accuracy(%)
With glasses	83.76
Night Without glasses	85.82
Night With glasses	79.45
Without glasses	88.89
With sunglasses	78.72
All	83.33

Mainly, the Android application has four pages as illustrated in Figure S2. The first page serves for logging in by using a username and password. Following the authentication, the user can start a new trip or access the information about the last five trips in the second page. If the user chooses a new trip, the application will start recording and displaying all information as described in the previous section. Then, it will send the collected data via the web service to the cloud server. In the fourth page, the front camera will capture and display the driver's face.

3.2. BLOCKCHAIN IOT SOLUTION FOR VEHICLES COMMUNICATION (DISV)

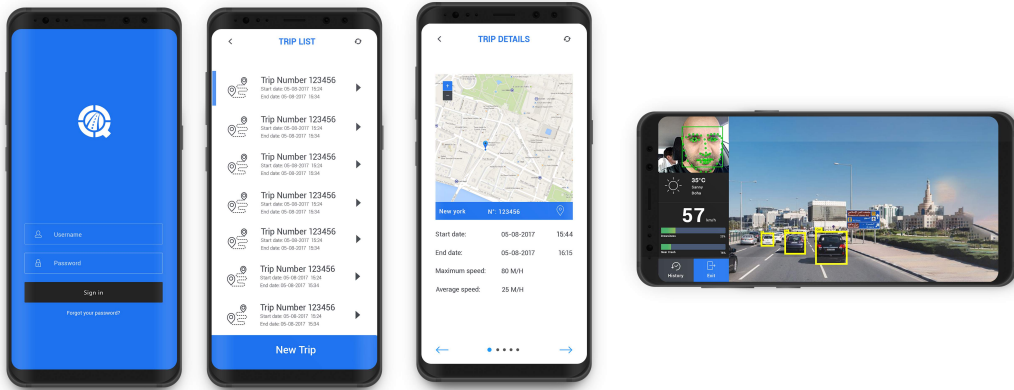


Figure 3.4: Screenshot of the four main pages of the Android application for Vehicles (AV).

3.2.2.2 Android Application for Infrastructure (AP)

The aim of this application is to simulate the role of IoT devices integrated into the roads such as radars, traffic lights, roadside electronic signs and others. Many additional options of the Android application, such as traffic jams, the speed of cars, and weather conditions can be added to the perception layer.

3.2.3 The Network Layer

The network layer establishes the connection between the servers and transmits, and processes the sensor data. The application can use either Wi-Fi or mobile internet (3G/3G+/4G/5G) to send the data to the server. This collection process uses the hybrid system to gather and store data locally before transmitting it them to the server. This technique is proven to be highly effective for data collection when the Internet connection is poor or unstable.

3.2.4 The Application Layer

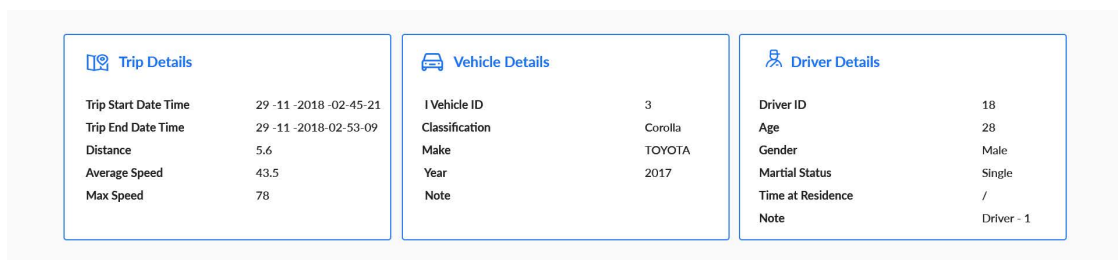
Regarding the application layer, it contains two principal compounds: Central cloud server and the communication system using a Blockchain Network.

3.2. BLOCKCHAIN IOT SOLUTION FOR VEHICLES COMMUNICATION (DISV)

3.2.4.1 Central Cloud Server

The central cloud delivers application-specific services to the end-user. It sends the collected data to the web services for processing and analysis before showing them to the end-user. The web service is a component of the application layer responsible for interaction between different components of the IoT solution, such as web site, database server, IoT devices, and embedded systems. Windows Communication Foundation by Microsoft is used to implement the web service based on the REST Architecture and JSON message format. In addition to collecting data from devices, it uses information about crashes from the General Directorate of Traffic at the Ministry of Interior as well as road conditions or any relevant data from other authorities. The data is available to the end-user through the website with direct access to the web services.

The web application is the interface that the researchers use to interact and query the recorded data. The website content, shown in Figure 3.5, displays demographic information about the driver's nationality, gender and age. It also includes information about the vehicle, including the model and date it was put into service.



The screenshot shows three panels of data for a trip. The first panel, 'Trip Details', lists start and end times, distance, average speed, and max speed. The second panel, 'Vehicle Details', lists vehicle ID, classification, make, year, and a note. The third panel, 'Driver Details', lists driver ID, age, gender, marital status, time at residence, and a note.

Trip Details	
Trip Start Date Time	29 -11 -2018 -02-45-21
Trip End Date Time	29 -11 -2018-02-53-09
Distance	5.6
Average Speed	43.5
Max Speed	78

Vehicle Details	
Vehicle ID	3
Classification	Corolla
Make	TOYOTA
Year	2017
Note	

Driver Details	
Driver ID	18
Age	28
Gender	Male
Marital Status	Single
Time at Residence	/
Note	Driver - 1

Figure 3.5: Screenshot of a real trip displaying information about the vehicle and the driver.

By using Google Maps, the website displays the tracked trip and the position of individual events, as well as the details of all recorded events as shown in Figures 3.6 and 3.7.

3.2. BLOCKCHAIN IOT SOLUTION FOR VEHICLES COMMUNICATION (DISV)

ID	Timestamp	Lat	Long	Speed	Acc-x	Acc-y	Acc-z	Roll-x	Yaw-y	PitchR-z	Note	Ima...
1379	29-11-2018-04-37:24...	0	0	0	-0.1728252...	10.228029	1.7525564	0.04941152...	0.21643016	-0.022136535		
1382	29-11-2018-04-37:24...	0	0	0	0.554548	3.7662342	9.481039	0.2999202	0.1809998	-0.064729495		
1383	29-11-2018-04-37:24...	0	0	21	0.009576807...	2.936001	8.772355	-0.3723637...	-0.13848254...	0.0020289204		
1384	29-11-2018-04-37:24...	25.3655651564576	51.495292284931886...	40	1.2142484...	7.9487495	6.3781533	-0.029470946...	0.021564154...	0.0062540383...		
1385	29-11-2018-04-37:24...	25.36326658741378	51.49502694220006	36	-3.3039982	4.1180267	8.523258	0.21834035	0.52797345	-0.142471899		
1386	29-11-2018-04-37:24...	25.368384622491314...	51.49496529861783	26	-5.48735	0.89944306	9.482152	0.56336247	-0.04746302...	-0.13942247		
1387	29-11-2018-04-37:24...	25.35897761243053...	51.49501861682879	1	-1.9536486	0.12499499	10.113708	-0.0015711454...	-4.269948E-4...	-1.4538951E-4		
1388	29-11-2018-04-37:24...	25.368381212639202...	51.49505328375402	31	-1.6663444	-0.16280572...	5.932855	-0.0034037412...	-0.00103786...	4.654793E-4		
1389	29-11-2018-04-37:24...	25.357393486269414...	51.49613288848309	43	6.397307	4.855441	6.588843	-3.491449E-4...	1.8387043E-4...	-0.0013671165...		
1390	29-11-2018-04-37:24...	25.355187336801244...	51.497921358975056...	41	0.23942018	7.4984396	9.883265	0.0014831808...	-0.00103786...	-1.4538951E-4		
1391	29-11-2018-04-37:24...	25.353242611988144...	51.49978034711291	40	1.292869	0.076614454...	9.835381	-9.602802E-4...	-0.001648752...	-1.4538951E-4		
1392	29-11-2018-04-37:24...	25.351381259607044...	51.50016929472988	41	1.1204864	0.641446	10.381259	-0.0011160509...	1.2217305E-5...	3.915646E-4		
1393	29-11-2018-04-37:24...	25.34986495222745	51.49762885570988	40	0.9197345	0.50757074	9.911995	7.165449E-4...	-0.0018203785...	-2.193006E-4		
1394	29-11-2018-04-37:24...	25.34877323684882...	51.49461538064034	45	0.2777774	-0.02873042...	10.553641	7.165449E-4...	-5.98448E-4	-0.0014410311...		

Figure 3.6: Screenshot of a real trip displaying the data recorded for every event.

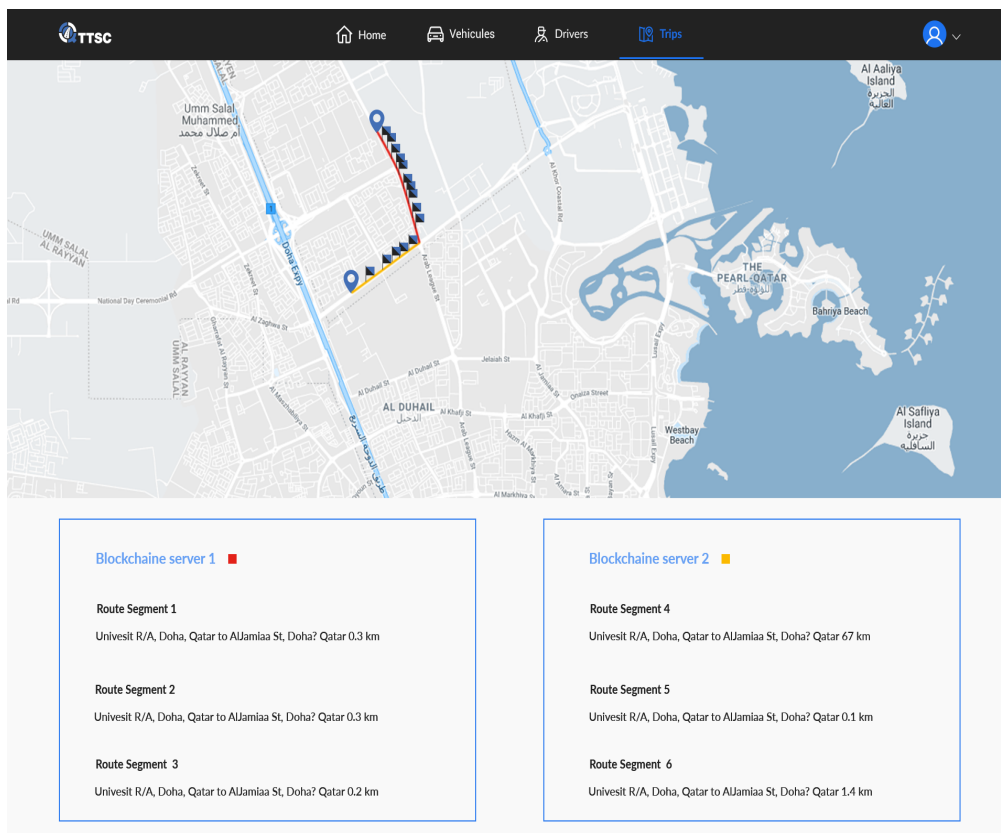


Figure 3.7: Screenshot of a real trip displaying the Blockchain layer for every road section.

3.2. BLOCKCHAIN IOT SOLUTION FOR VEHICLES COMMUNICATION (DISV)

3.2.4.2 The Blockchain Layer

3.2.4.2.1 Blockchain Layer Overview The Blockchain layer manages communication between cars. In each separate time slot, the car sends collected data to the central server via a web service. The data includes the current location and the status of the connection to one of the existing Blockchain layers. Subsequently, the central server invites nearby IoT devices to establish communication by an available Blockchain cloud. The communication is established after accepting the invitation. As illustrated in Figure 3.9, each road section contains a Blockchain layer, which sends the messages to the connected IoT devices.

3.2.4.2.2 System In- Depth The Blockchain layer and the Android application jointly create decentralized applications (Dapp, dApp, or DApp). More precisely, decentralized applications represent distributed Internet apps run on a decentralized P2P network (Blockchain).

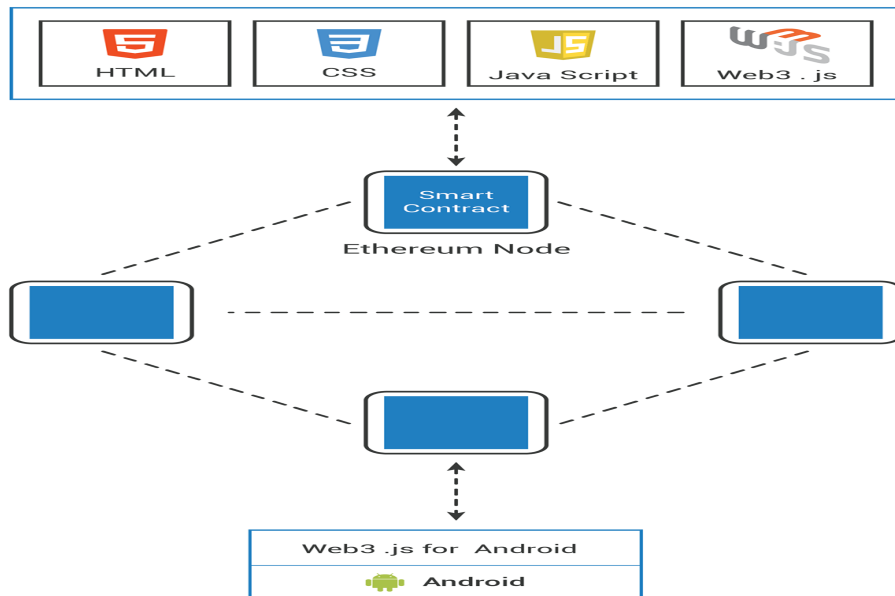


Figure 3.8: The architecture of decentralized applications (Dapp).

Their code is an open source that is publicly open, and accordingly, it can be accessed and customized. Dapp applications do not rely on a central server like standard apps, as illustrated in Figure 3.8. The Blockchain layer is the back-end of the decentralized applications, whereas the android application is the front-end part. The mobile app calls functions of the smart contract deployed in every

3.2. BLOCKCHAIN IOT SOLUTION FOR VEHICLES COMMUNICATION (DISV)

Ethereum node in order to send a message via the Blockchain network. The communication passes through as a wrapper between the mobile and node-endpoint. This study uses one of the most reliable frameworks - the Web3.Js for Android framework. Its smart contract has two primary functions. The first function, “setMessage”, is in charge of publishing a new message on the Blockchain network in the amount of ETH the sender is willing to pay per unit of gas to mine the message. The second function, “GetMessage”, enables the device connected to the Blockchain network to read the existing data. Table 3.3 shows the format of the sent message. The message is composed of the following elements:

Table 3.3: Model of the sent message in the Blockchain network.

Model	Example
<pre>{ Sender : , TypeSender : , Time : , FinishTime , Message , TypeMessage: Position , }</pre>	<pre>{ ■s": "Toyota , 404551 , white", ■ts": "Car", ■t": " 2018-10-13 19:43:16", ■tf": " 2018-10-13 19:53:16", ■m": "Alert Drowsy driver", ■tm": "3", ■p": " 25.333091, 51.467223", }</pre>

The message is composed of many elements:

- “s” field presents the information about the sender such as brand, car matriculation number, and color of the car.
- “ts” field defines the type of sender message that can be sent from a car, pedestrian or infrastructure.
- “t” field contains the time of sending the message. This helps the Android application to decide if the message is new or old. Therefore, if the mobile receives the message late, the notification alert will not be displayed.
- “tf” field contains the time when the message should be removed from the smart contract. Every time any participant adds a new message, it must delete the obsolete messages. Thus, the smart contract becomes lighter and remarkably reduces the execution time, mining time, cost

3.2. BLOCKCHAIN IOT SOLUTION FOR VEHICLES COMMUNICATION (DISV)

and energy. The finish time value is the result of summing between the start time and duration of the message. The availability of the message in the smart contract is detailed in Table 3.4.

Table 3.4: Urgency level of the message sent through the Blockchain Network.

Urgent Level	Duration of Message	Example
0	10 min	Drowsy driver or bad driver behavior
1	1 h	Streets crowded
2	6 h	Temporary Closed Roads
3	12 h	Maintenance work

- “m” field includes the content of the message. For example : “Be careful of a drowsy driver, road crash in Khalifa Street”.
- “mt” field defines the priority of the message as explained in Table 3.5; there are three message types: information, warning, incident.
- “p” field contains the position of the sender and will not be displayed in case of delay in sending the message and when the sender becomes far from the incident point.

There are certain limits regarding the use of Blockchain for establishing communication between vehicles and other participants in the transportation system. The main concern is the time needed to update the transaction into a Blockchain. The DISV cannot be regarded as a real-time application due to the time needed for updating the smart contract. Therefore, several measures are proposed in order to reduce the content of the smart contract and consequently minimize the time required. First, instead of having one Blockchain layer in charge of the communication in a larger region, separate Blockchain layers within smaller areas would facilitate the communication of a low number of vehicles. Second, following the competition of their time, all messages should be removed from the system (see Table 3.4). Third, as elaborated in Table 3.3, the format of messages should be minimal to increase the transaction speed. Furthermore, the table 3.4 illustrates the duration of every message based on Urgent level. In fact the message received by “SetMessage” function in smart contract will

3.2. BLOCKCHAIN IOT SOLUTION FOR VEHICLES COMMUNICATION (DISV)

stored in Patricia Trie data storage on Ethereum Blockchain [217]. Once they become obsolete, the Smart contract removes the message after the central data receives a copy of it.

Table 3.5: Type of message sent through the Blockchain Network.

Type	Message
Information message	i.e., Informative messages from the Ministry of Interior. It is displayed only in the message page of the Android application.
Warning message	i.e., Information about traffic signal not working. It is displayed as a pop-up message.
Incident message	i.e., A drowsy driver, critical zone or extreme weather condition. It is displayed as a pop-up message with an alert sound to get the driver's attention.

3.2.5 Nominal Scenario

In this section, the nominal scenario of communication between IoT devices is described. The sequence diagram in Figure 3.9 shows that the nominal scenario is divided into two sub-processes; registration and messaging. In the registration step, for every time slot (15 s), the IoT device sends the collected data to the central server via the Internet (1). The central server saves the collected data in the database server (2). In addition, the server looks for the IoT devices nearby places such as road section, roundabouts, traffic signals, or others (3). Subsequently, an invitation is sent to the devices in the same location to communicate through one of the available Blockchain layers (4). After accepting the invitation, (5) the second sub-process begins. In the messaging step, the IoT devices are now connected to each other and can share data between them. The IoT devices send the message via the Blockchain network. Following the mining process (6), the message is added to the smart contract so that every device connected to this server Blockchain layer can receive it (7).

3.2. BLOCKCHAIN IOT SOLUTION FOR VEHICLES COMMUNICATION (DISV)

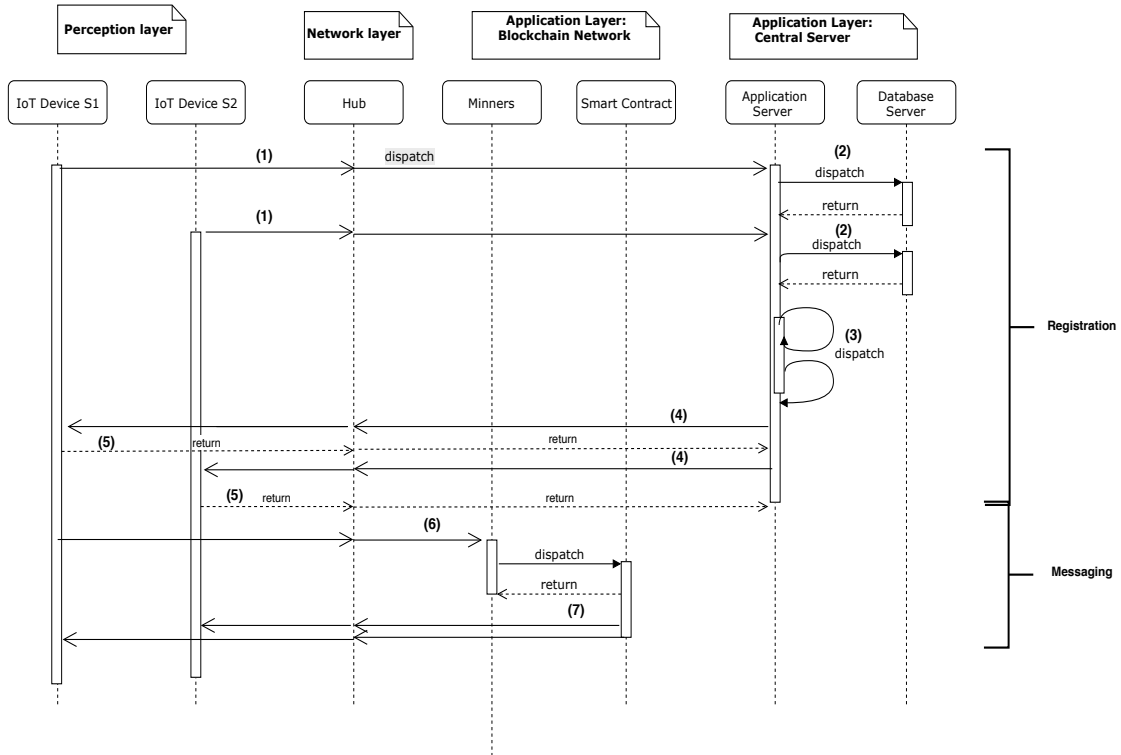


Figure 3.9: Sequence diagram of nominal scenario of communication between Internet of things (IoT) devices.

3.2.6 Formal Testing Framework

3.2.6.1 Test Generation Principle

Our generation procedure is inspired by the work of [218]. A test case may be considered as a tree. The nodes of the test tree may be seen as collections of states S of the model of the “System Under Test” (SUT). The adopted test generation procedure is in charge of extending the test tree by defining successors to an every leaf node, as shown in Figure 3.10. For every *non-acceptable* output a_i the test tree moves to *fail* and for every acceptable output b_i , the test tree moves to a new node which corresponds to the set of states that the system can reach after producing b_i . The tester may also decide to emit a valid input c from the current node (dashed arrow).

3.2. BLOCKCHAIN IOT SOLUTION FOR VEHICLES COMMUNICATION (DISV)

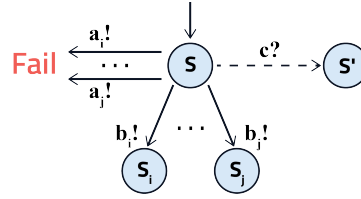


Figure 3.10: Test generation principle

3.2.6.2 Combining Functional and Load Aspects

At this level, our goal is to combine load and functional aspects in our modelling since our system is made of a number of interacting and concurrent components. For this purpose, we adopt an extended variant of Timed Automata equipped with integer shared variables.

As illustrated in Figure 3.11, the used integer variable of the proposed timed automaton corresponds to the number running instances of the considered system. In this example, we demonstrate how the answer time to generate an action b may vary according to the number of running instances.

3.2.6.3 Testing Security Aspects Using Attack Trees

In the literature, “Attack Trees” [219, 220] are used to assess the security of critical systems. They allow to represent graphically the strategy of a given attacker. An example of an AT is proposed in Figure 3.12 [219]. In this example, the considered attacker aims at cracking the password of some protected files.

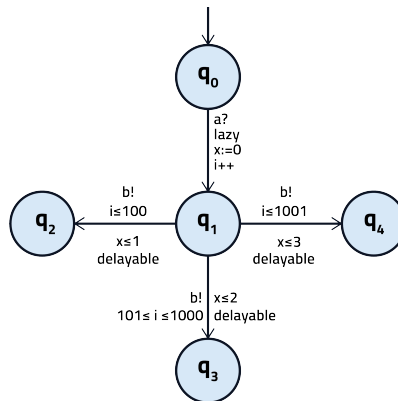


Figure 3.11: An example showing how the response time of the system under test varies regarding the current load level.

In general, the root of an attack tree corresponds to the global goal of the attacker and the leaves of the tree correspond to basic attack steps the attacker needs to combine in order to achieve its global goal. Internal nodes correspond to intermediary sub-goals. The attack tree has two types of gates namely AND-Gates and OR-Gates. On the first hand, an AND-gate means that in order to fulfill the goal a parent-node all sub-goals of children-nodes of the considered node have to be achieved. On the other hand, an OR-Gate means that the goal of a parent-node can be achieved by fulfilling the sub-goal of only one of its children-nodes.

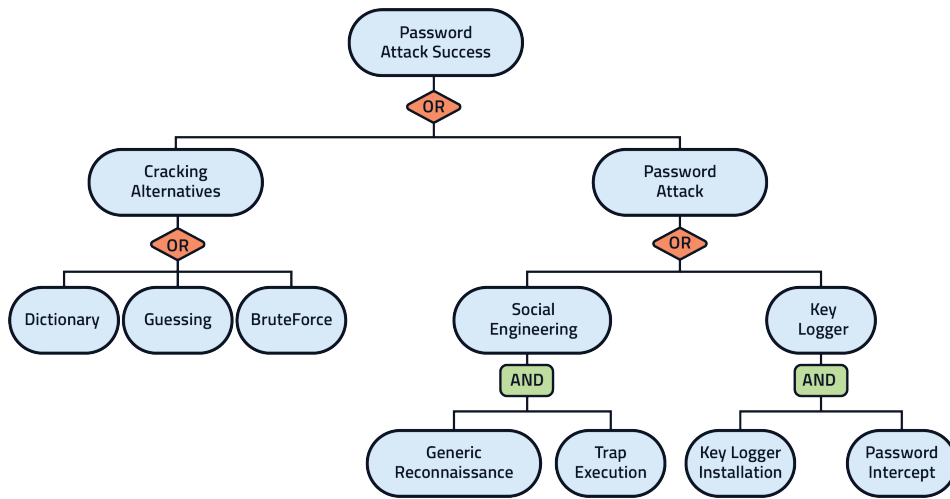


Figure 3.12: Example of an Attack Tree

After defining the attack tree modelling the behavior of the attacker, the second step consists in transforming the obtained tree into a network of Timed Automata which will serve as an input for our test generation procedure. The proposed transformation is inspired by the transformation proposed in [221].

3.3 Blockchain IoV Solution for payment (PSEV)

This proposed solution provides secure communication between vehicles and other actors in transportation systems and allows the proceedings of payments. It attempts to overcome limitations such as long execution time, and accordingly, improves performance. Since it is based on an IoT architecture, the proposed solution consists of three layers: the perception, the network, and the application layers, as illustrated in Table 3.6 and described below:

3.3. BLOCKCHAIN IOV SOLUTION FOR PAYMENT (PSEV)

1. Perception Layer - This layer consists of three developed solutions in the form of Android applications, namely Android Auto Application for Vehicles, Android Application for Parking Space Renters, and Android Application for the Parking IoT System. Using these three applications, the users can establish communication, manage their own parking space, and conduct payments.
2. Network Layer - This layer uses wireless networks to connect the application layer to other IoT devices, such as parking and vehicle sensors.
3. Application Layer - This layer hosts the server and applications responsible for the communication and payment system. This layer meets the particular needs of the management system of the devices included in the Blockchain solution.

Table 3.6: The main features of the developed PSEV solution.

Layers	Developed Solution	Main Features
Perception layer	Android Auto Application for Vehicles	This auto-based application is the interface in the car to be employed by the user. This console is used to send and receive messages, and search, navigate, and conduct parking payment when parking is required by the driver.
	Android Application for Parking Space Renters	This Android application allows the building/parking space owners to manage and set specific time slots in which their parking spaces are available for rent to the public when not in use.
	Android Application for the Parking IoT System	This is an embedded application in the IoT parking payment machine that detects cars in the Blockchain that are nearby and grants access to parking lots for cars that submitted a request.
Network layer		Connects the application in the perception layer to other network devices, servers, and smart things.
Application layer	Blockchain Application	Manages payment and establishes communication between vehicles and other actors in the transportation system.
	Central Cloud Server	The central cloud server serves as a host of the management and administrative software. Also, it hosts applications that run the service.

3.3. BLOCKCHAIN IOV SOLUTION FOR PAYMENT (PSEV)

The proposed architecture as shown in Figure S3 will be explained in more detail in the following section.

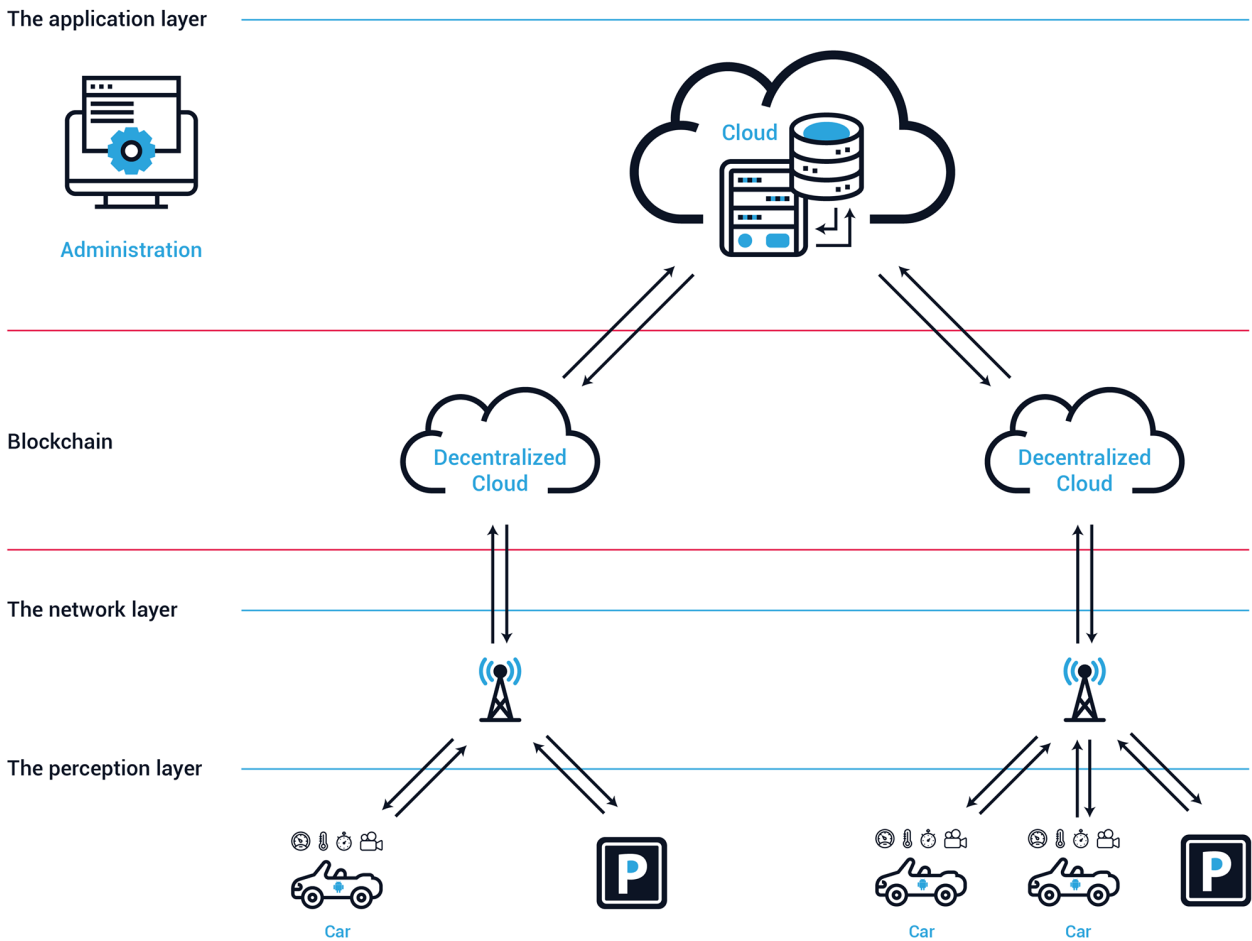


Figure 3.13: The developed Internet of Vehicle architecture PSEV.

3.3.1 The Perception Layer

This layer incorporates different payment portals that establish the interaction. The proposed solution includes three separate applications for vehicles and parking infrastructure required to ensure secure communication and payment.

3.3. BLOCKCHAIN IOV SOLUTION FOR PAYMENT (PSEV)

3.3.1.1 Android Auto Application for Vehicles

This is the car interface employed by the user for sending and receiving messages and searching, navigating, and paying for parking. In addition, this application detects the proximity of the car to the parking payment station to initiate the Blockchain-based transaction. The content of the Android app is illustrated in Figure S4 and Figure S5.



Figure 3.14: Screenshot of the Android application interface inside the vehicle.

As it is shown, the interface is the login page that appears the application turns on. Following the authentication, the screen shows the map with currently available parking locations. Once the user selects an appropriate location, the app provides the navigation information. As the user reaches the prescribed parking area, the application shows the list of parking spots available at the selected

3.3. BLOCKCHAIN IOV SOLUTION FOR PAYMENT (PSEV)

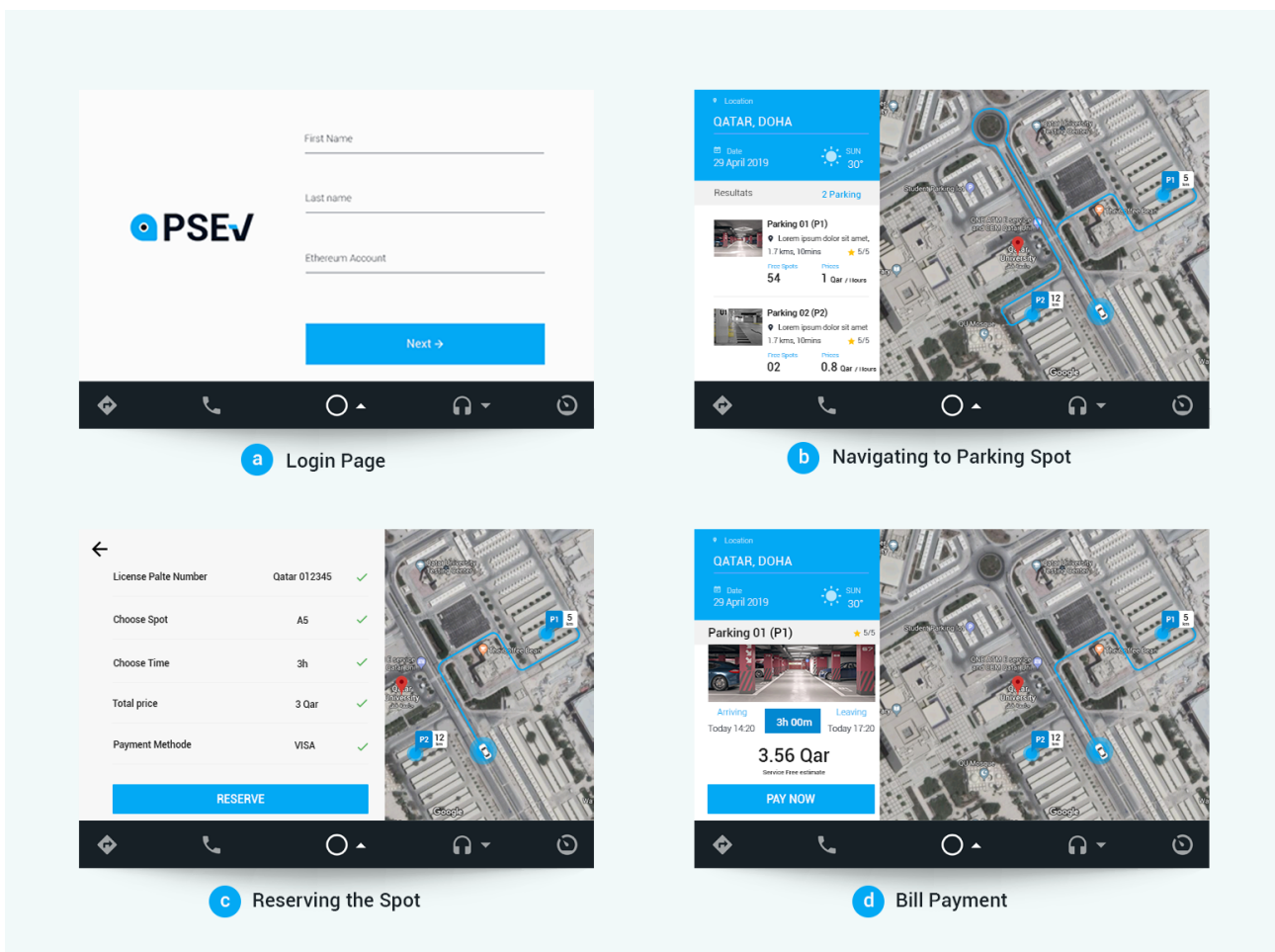


Figure 3.15: Screenshot of the primary navigation pages of the Android application.

3.3. BLOCKCHAIN IOV SOLUTION FOR PAYMENT (PSEV)

location and their time slots. The user selects the most suitable slot according to his needs. Once selected, the application initiates the timer and displays the cost. When leaving the parking lot, the driver clicks “STOP” on the timer and the application shows the summary of the transaction. Besides, the user can access to the history tab where to check all previous transactions and their details.

3.3.1.2 Android Application for Parking Space Renters

This Android application enables the building/parking space owners to manage and set time slots in which their parking spaces are available for rent to the public, when not in use. Hence, the parking owners can generate revenues from the unused parking slots. This application has mainly three pages. The first one is the login page for authentication. Second, a parking spot configuration page allows the user to define the time slots in which his parking space is available for rent. The renters can access the lists of previous transactions, if any, and the total amount earned for the given parking space. Finally, a settings page serves for account management and administrative tasks.

3.3.1.3 Android Application for the Parking IoT System

This is an embedded application in the IoT parking payment machine that will detect cars in the Blockchain that are nearby and grant access to parking lots for cars that have requested them. This payment portal will not take care only of the authentication, time slot management but also the payment transaction between the car and the parking space provider. This application can play the role of the parking access machine that guarantees physical access to the parking lot. All the above applications are players in the Blockchain system, and the transactions and messages sent are secured by the Blockchain contracts.

3.3.2 The Network Layer

The purpose of the network layer is to establish the connection between parking payment portals located in the perception layer (Android Auto Application for Vehicles, Android Application for Parking Space Renters, and Android Application for the Parking IoT system), smart cars, and the central cloud server in the Blockchain sub-layer. This layer transmits and processes the IoT data securely through WiFi or mobile internet (3G/3G+/4G/5G).

3.3.3 The Application Layer

Central cloud server and communication system are the main components of this layer. They operate using the Blockchain network.

3.3.3.1 Central Cloud Server

The management and administrative application are hosted on the central cloud server. The cloud monitors the systems and functioning of the infrastructure in the role of an admin. Thus, it has a function of hosting and administrating services of all payment portals and cars participating in the system. The user employs the Android Auto Application for Vehicles to purchase the ticket. Next, the central server invites the IoT devices in the same area to communicate with each other via one of the existing Blockchain cloud instances. More precisely, IoT devices receive an invitation to communicate via the Blockchain sub-layer. A web application, shown in Figure 3.16, resides in the central cloud. It serves as a tool in the application layer employed to establish interaction among various elements of the IoT solution, including the administrative web site, a database server of the devices that interact with the Blockchain, and the embedded systems at the payment portals. Windows Azure cloud service with REST API calls has been utilized to host this service. On this web application, several scientific tools that can analyze the traffic flow at different parking spots are also created in order to study and understand the parking demand requirements in different locations of the city. Thereby, this serves as a central administrative platform to be used by administrator to troubleshoot the system if necessary and ensure smooth functioning.

3.3.3.2 Blockchain sub-layer

The Blockchain sub-layer is responsible for facilitating communication among various elements and road users of smart cities and the payments between cars and parking providers. More precisely, every time a user intends to conduct a parking payment, this transaction is broadcasted to the Blockchain sub-layer and validated. Completed transactions become a part of the block, which makes them valid and immutable. The Blockchain sub-layer and the Android app are merged to generate decentralized apps (Dapp,dApp, or DApp). Decentralized apps are distributed Internet apps that operate on a decentralized P2P network such as Blockchain. The Android app is the front-end, while the Blockchain

3.3. BLOCKCHAIN IOV SOLUTION FOR PAYMENT (PSEV)

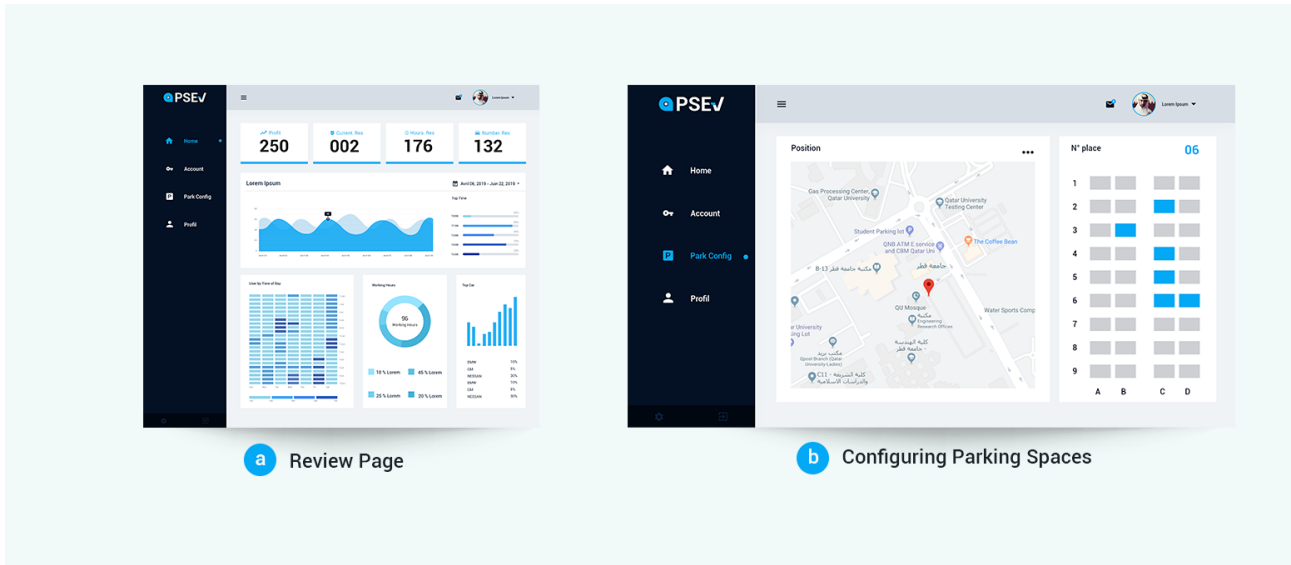


Figure 3.16: Screenshot of the web portal of the central cloud.

sub-layer is the back-end parts of the decentralized apps. The smart contract functions are deployed in all Ethereum nodes, and the mobile app uses the Blockchain for sending a message. The mobile and node-endpoints establish communication. This solution employs exceptionally reliable Web3.Js for Android framework.

The Blockchain sub-layer contains the following system:

3.3.3.2.1 Access Management System The Access Management System enables the identification and authentication of different participants of intelligent transportation systems in order to exchange data. The following modules are located in the Blockchain sub-layer in the Access Management System:

- **Vehicle Management Module:** The Vehicle Management Contract is a crucial element of this module. It enables adding, modifying, and deleting a vehicle.
- **Parking Management Module:** The Parking Management contract is also an important element of this module. It enables adding, modifying, and suppressing a parking in the system.
- **Participant Management Module:** The Participant Management Contract enables adding, modifying, and suppressing another participant to the intelligent transportation system that can send

3.3. BLOCKCHAIN IOV SOLUTION FOR PAYMENT (PSEV)

messages to the system including roadside, electronic signs, traffic lights, and radars.

3.3.3.2.2 Parking Payment Management System This system allows the management of the parking. It contains smart contract that enables the providers to sell tickets and get numbers for the available tickets. In addition, it contains a smart contract that enables clients or vehicles to buy available tickets.

3.3.3.2.3 Communication Management System The smart contract of the Communication Management System has two primary functions. “Send Message” function transmits a new message on the Blockchain network. This action is based on the ETH amount the sender is willing to give per unit of gas for mining the message. “Read Message” function allows the participant to access the data through the devices that is a part of the Blockchain network.

3.3.4 Nominal Scenarios

3.3.4.1 Parking Management System

In this section, two nominal cases in which the system will be utilized are shown below:

1. Requesting for Parking
2. Providing parking as a supplier

3.3.4.1.1 Requesting for Parking The procedures involved in the transaction are listed as below: At the moment the driver requires parking, he taps on the “Find Parking” icon. The application will send this request to the central server, the server will return the list of different locations nearby that are available with their respective prices. The application will display the nearby parking spots on the screen. The user clicks the preferable available parking spot. The application turns on the navigation to the said destination, as shown in Figure 3.17.

Once at the destination, the payment portal will authenticate the car using the Blockchain protocols and verify with the server to make sure the customer is enrolled in the Blockchain and is a valid customer. Next, the user is provided with the parking spaces inside the parking lot, with the list of

3.3. BLOCKCHAIN IOV SOLUTION FOR PAYMENT (PSEV)

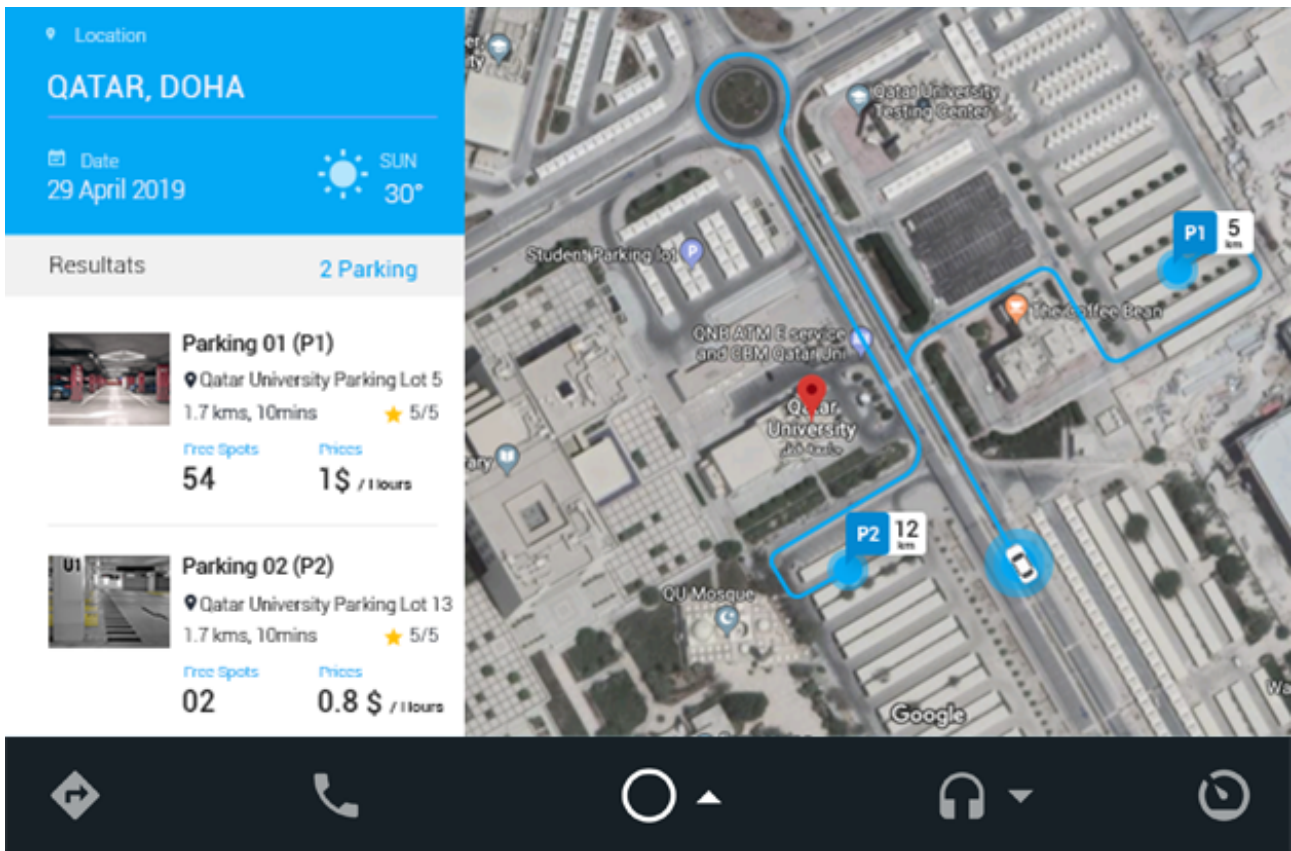


Figure 3.17: Screenshot of Android Application for Vehicles displaying requesting for parking.

prices and time slots available. The user selects the required one and the reservation is confirmed as shown in Figure 3.18. The payment portal, which is the control unit that grants access to the physical location, opens the gate and allows the car to enter. The user looks for the selected parking space and parks his car at the designated location. Once parked, the application starts the timer and display the time as well as the fees incurred on the screen.

Once the user wishes to leave the parking lot, he drives out of the parking lot, and the payment portal notices that the car is exiting. The application then displays the total time and cost incurred. The Android Auto Application for Vehicles will pay the parking lot machine automatically for the above said amount over the Blockchain. This transaction is then verified and noted in the Blockchain channel.

3.3. BLOCKCHAIN IOV SOLUTION FOR PAYMENT (PSEV)

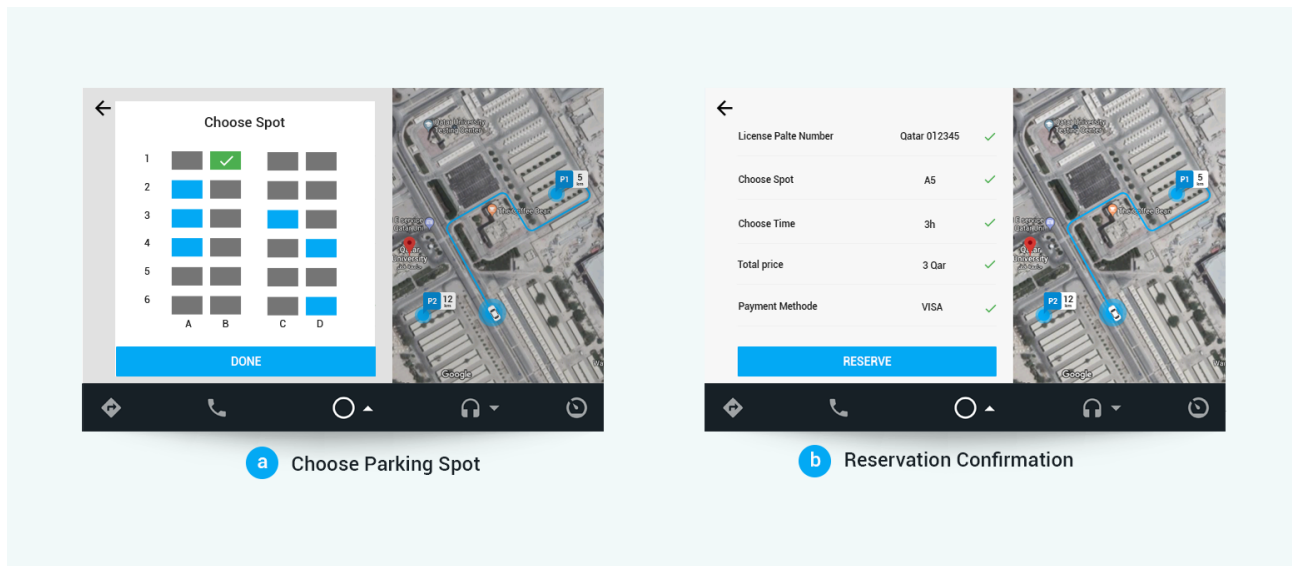


Figure 3.18: Screenshot of developed web application.

3.3.4.1.2 Providing parking as a supplier The nominal procedure involved in a user renting out his parking space is as follows: The user logs into the Android Auto Application for Vehicles via the login page. The user selects the parking lot from his list, sets the time slots appropriately as per his requirements, and submits the request for broadcasting to the Blockchain. The server receives that information and make those parking spots available for the set duration to all drivers using the service.

3.3.4.2 Communication Management System

We used the following scenario: sending alert messages in case of drowsiness to develop and test the prototype of PSEV-Communication. In the case of a drowsy driver, the cars nearby are alerted by a message transmitted through the Blockchain. We included the factor of driver drowsiness, considering that it is one of the leading causes of motor vehicle crashes that was confirmed by a study [222] conducted by the AAA Foundation for Traffic Safety. Essentially, this report implied that over 5,000 Americans lost their lives as a result of sleep-related vehicular crashes. The car sends to the Blockchain sub-layer a message that contains essential information, such as who is the sender, time, content of message, and the position. After the mining process is completed, the message becomes a part of the smart contract, and accordingly, any device that is connected to this server Blockchain sub-layer is able to access it as illustrated in Figure 3.19. The central server receives a copy of the message in the

3.3. BLOCKCHAIN IOV SOLUTION FOR PAYMENT (PSEV)

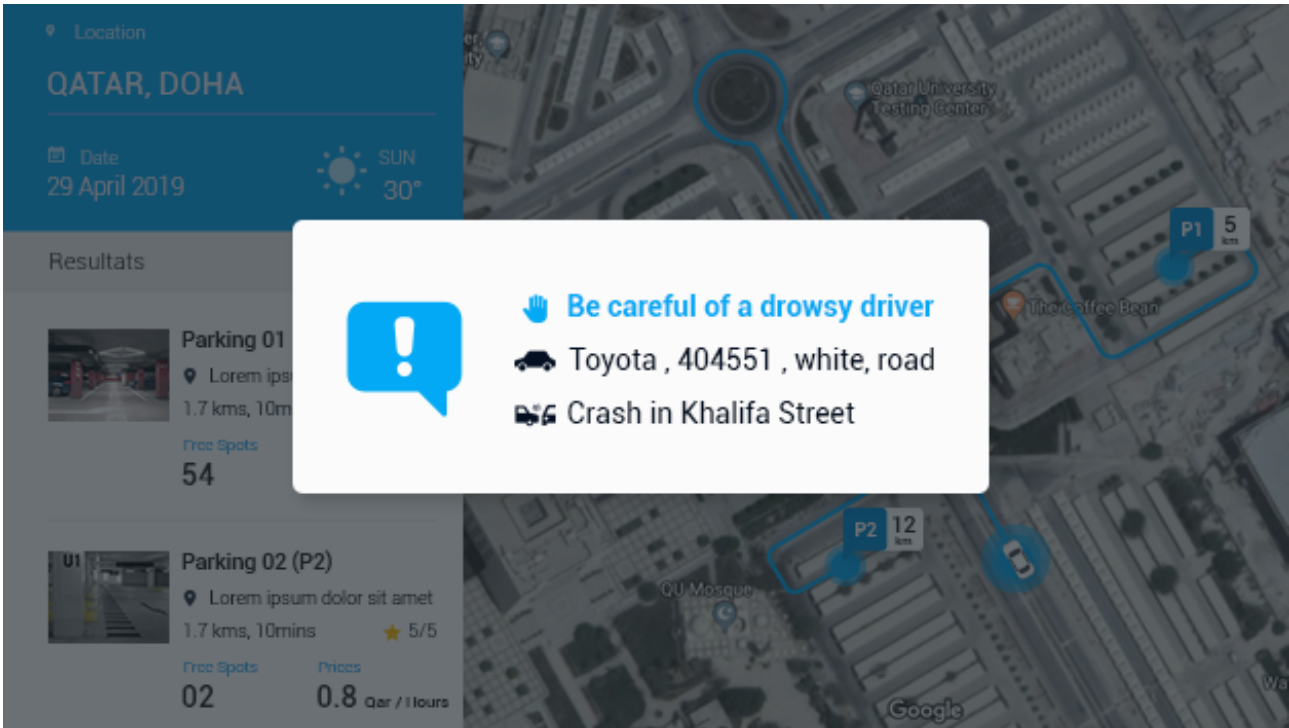


Figure 3.19: Screenshot of a message received to alert the driver.

Blockchain.

3.3.5 Formal Testing Framework

3.3.5.1 Model-Based Testing

“Model-Based Testing” (MBT) [223–227] is a methodology where the system of interest is described by a mathematical model which encodes the behavior of the considered system. This methodology consists of using this mathematical model to compute abstract test scenarios. These sequences of concrete sequences are then transformed into concrete test sequences which are executed on the considered system under test. The verdict of this testing activity is provided by comparing the observed outputs from the system with the outputs generated by the model.

Timed Automata (TA) [228,229] represents an expressive and simple tool for describing the behavior of computer systems. It combines continuous and discrete mechanisms. TA may be represented as finite graphs enriched with a finite set of clocks defined as real entities whose value progresses

3.3. BLOCKCHAIN IOV SOLUTION FOR PAYMENT (PSEV)

continuously over time. Figure 3.20 illustrates a proposed example of a TA which has five nodes, four transitions, three actions, and one clock.

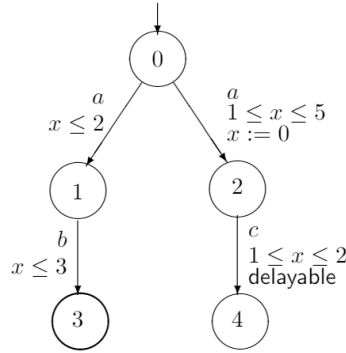


Figure 3.20: A Timed Automata with 5 states, 4 transitions, 3 actions and 1 clock.

A test case is an interactive scenario taking part between the “System Under Test” (SUT) and the tester. The tester provides input actions to the SUT and collects generated corresponding output actions. Subsequently, the tester verifies whether the produced output sequences are acceptable or not and emits correspondingly either *fail* or *pass* verdicts. Typically, it is possible to categorize three kinds of testers:

- TA Testers: represented as Timed Automata.
- On-the-Fly Analog Testers: able to measure time with precision.
- Digital Testers: measure time with less precision.

The suggested generation procedure is drawn upon the work of [218]. A test case may be considered as a tree. The nodes of the test tree may be seen as collections of states S of the model of the SUT. The adopted test generation procedure is in charge of extending the test tree by defining successors to an every leaf node, as shown in Figure 3.10. The tester may also decide to emit a valid input c from the current node (dashed arrow). The test generation procedure to be adopted is shown in Figure 3.22. More details about this procedure can be found in [223].

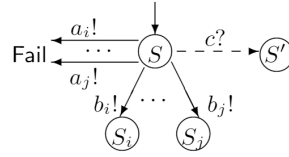


Figure 3.21: Test generation principle

```

1   $S \leftarrow \text{tsucc}(\{s_0^{As}\}, 0);$ 
2  while(true)
3     $x \leftarrow 0;$ 
4    await(output  $b$  is received at  $x < T$  or  $x = T$ )
5    if ( $b$  received at  $x$ )
6       $S \leftarrow \text{dsucc}(\text{tsucc}(S, x), b);$ 
7    else
8       $S \leftarrow \text{tsucc}(S, T);$ 
9    endif;
10   if ( $S = \emptyset$ )
11     announce Fail;
12     exit;
13   endif;
14   if ( $\text{valid\_inputs}(S) \neq \emptyset$ )
15      $i \leftarrow \text{pick}(\{0, 1\});$ 
16   endif;
17   if ( $i = 0$ )
18      $a \leftarrow \text{pick}(\text{valid\_inputs}(S));$ 
19      $S \leftarrow \text{dsucc}(S, a);$ 
20   endif;
21 endwhile;
```

Figure 3.22: Test generation procedure

3.3.5.2 Combining Functional and Load Aspects

At this level, our goal is to combine load and functional aspects [230–232] in our modeling since our system is made of a number of interacting and concurrent components. For this purpose, we adopted an extended variant of Timed Automata equipped with integer shared variables. As shown in Figure 3.23, the used integer variable of the proposed timed automaton corresponds to the number running of instances of the considered system. By this example, we demonstrated how system under the test may produce complete distinct behaviors corresponding to different load levels.

3.3.5.3 Improving Formal Testing Methods

In this section, several techniques which may be used for improving formal MBT methods are explained.

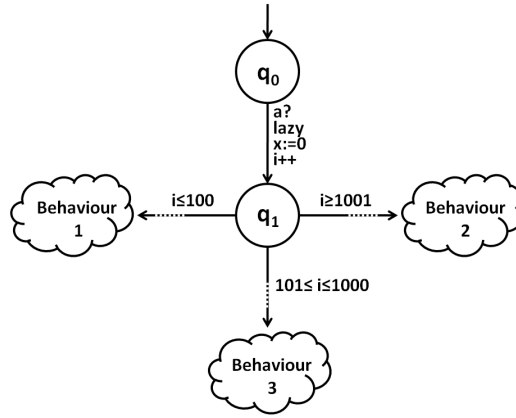


Figure 3.23: A System Under Test which has three possible behaviors corresponding to different load levels.

- “State Identification”: The state identification problems [233, 234] were initially introduced for the case of finite state machines (FSMs). The solution for this problem consists in identifying either the initial or the final state of the considered machines.
- “Coverage Techniques”: Several coverage techniques, such as statement coverage and branch coverage, can be used in the testing field [235]. Similarly, for timed systems existing methodologies [236] can be used.
- “Producing Timed Automata Testers”: This technique allows to produce a deterministic approximation of the tester in the form of a timed automaton using appropriate algorithms and heuristics such as the ones presented in [237–239].
- “Diminishing the Size of Digital Testers”: Digital testers may become very large since they may contain very long sequences of *tick* actions. A possible solution to tackle this problem extending testers with more sophisticated variables and data structures [236].
- “Refinement Techniques”: These techniques consist of converting high-level symbols into sequences of lower-level symbols. In [240], the authors proposed a refinement based methodology for testing timed systems.
- “Exploiting Reversible Rules”: This method [241, 242] allows the collapse of the subgraphs of the state graph into abstract states called progenitors.

- “Removing Functional Dependencies”: In [243] functional dependency is detected using Craig interpolation methods SAT solving. In [244], the authors detected functional dependencies from transition functions and not from the computation of the reachable states.
- “Data Independence Identification”: It [245, 246] can be used in the case when the designer of the system under verification identifies the fact that the behavior of the system is independent of some particular inputs.
- “Symmetry Identification” [247, 248] is a method based on using symmetries occurring during the execution of the system, for the purpose of minimizing the considered state space.
- “Abstraction”: The authors of [249] focused on verifying cyber-physical systems. Fundamentally, they applied specific transformations to remove details irrelevant to the properties of interest.

3.3.5.4 Testing Dynamic Adaptations

In this section, we deal with run-time testing after behavioral adaptations [250–253]. We presume that after a dynamic adaptation occurs the model of the considered system can change either completely or partially. Accordingly, the collection of available test scenarios must be modified either by generating new test cases or by changing old ones. To achieve that goal, a four-step strategy is followed as described below:

- “Model Differentiation”: compares the new and old models of the SUT for detecting differences and similarities between them;
- “Old Tests Classification”: divides old test cases into three groups (1. Tests no longer valid, 2. Tests partially valid and 3. Tests still valid);
- “Test Update and Generation”: updates partially valid test cases and generates novel tests which cover the new behaviors of the SUT;
- “TTCN-3 Transformation”: translates abstract test cases into TTCN-3.

3.3.5.5 Test Isolation

It is necessary to adopt different isolation strategies during the test execution step in order to prevent interference between testing and business behaviors [250]:

- “Blocking-based strategy”: prohibits instantly the SUT from receiving any queries from its environment, except from the test component to which it is associated.
- “Cloning-based strategy”: produces copies of the SUT and then tests the generated clones instead of the SUT.
- “Tagging-based strategy”: tags test queries with special tags for differentiating them from business queries.
- “Aspect-based strategy”: equips the SUT components with aspects that provide testing facilities [254].
- “BIT-based strategy”: is a built-in-testing strategy where the SUT is equipped with an interface which accepts both test and business queries in parallel.

3.3.5.6 Optimization of Testers Placement

This problem is drawn upon fog computing approaches [255, 256] and by previous contributions by [257, 258]. It consists of allocating the set of testers on the different computational nodes of the SUT in an optimal manner under several types of constraints as mentioned below.

3.3.5.6.1 Different Types of Constraints

- “Application Constraints”: In [259], the demand rate of the applications was considered.
- “Energy Constraints”: For instance in [256], the fog nodes were characterized with their energy capacities.
- “Network Constraints”: In [255] only latency constraint was considered.
- “Node Constraints”: For example, the authors of [260] took into account storage constraints.

3.3.5.6.2 Objective Functions

- “QoS-assurance”: In [255], the chosen QoS requirement corresponded to reaching execution times that are smaller than the application deadlines.
- “Cost”: In [261] the authors aimed at minimizing the total cost of deploying the considered applications by allocating them computational nodes with minimal costs.
- “Migrations”: In [262], the migrations number was optimized by reducing the network use without impacting its latency.
- “Execution Time and Network Delay”: This objective function was adopted by the authors of [263].
- “Energy”: The authors of [256] considered a linear objective measure of energy consumption.

3.3.5.6.3 Algorithms

- “Complex Networks”: Filiposka et al. [264] used network science theory to investigate the placement problem.
- “Deep Learning”: In [265], the authors exploited modern learning techniques for solving the placement problem.
- “Genetic Algorithms”: In [266], the authors proposed parallel genetic algorithms in order to deal with placement problem.
- “Game Theory”: In [267], the placement problem was encoded as a pair of games.
- “Mathematical Programming”: Several works like [260] solved the placement problem using this mathematical method.
- “Dynamic Programming”: In [268] the placement problem was modelled as a multidimensional knapsack problem.
- “Search-based Algorithms”: In [269] an algorithm was proposed to find a placement scenario for The Internet of Things applications.

3.4 Conclusion

This chapter introduced a new Decentralized IoT solution for Vehicle communication (DISV), which consists of three primary layers. We developed this solution to investigate the options of applying Blockchain for communication in the IoV. The nominal scenario of communication between IoT devices are introduced. The proposed testing approach uses Attack Trees and Timed Automata to evaluate functional, load and security aspects. An optimization phase for testers placement drawn upon fog computing is also proposed. We deployed a prototype of the smart contract on the Testnet of Ethereum.

Moreover, this chapter introduces the proposed innovative Blockchain framework for vehicle communication and parking payment (PSEV). It consists of three primary layers in charge of communication and payments in the IoV using Blockchain. Furthermore, nominal cases for using the system are discussed, as well as Model-Based Testing techniques, which represent deriving test suites from an adopted formal model, performing them, and assessing the correctness. This model combines functional and load aspects. The solution is based on the smart contract prototype on the TESTNET Ropsten of Ethereum.

Chapter 4

Performance evaluation

Contents

4.1	Introduction	102
4.2	Blockchain IoV Solution for Vehicles communication (DISV)	102
4.2.1	Costs	103
4.2.2	Execution Time	104
4.2.3	Memory and Power Consumption	106
4.2.4	Availability	107
4.2.5	Integrity	108
4.2.6	Consistency	109
4.2.7	Confidentiality	110
4.2.8	Immutability	110
4.2.9	Security	111
4.3	Blockchain IoV Solution for payment (PSEV)	113
4.3.1	Cost	114
4.3.2	Execution Time	115
4.3.3	Integrity	117
4.3.4	Consistency	118
4.3.5	Confidentiality	119
4.3.6	Immutability	119
4.3.7	Memory and Power Consumption	120
4.3.8	Security	122
4.4	Discussion	127
4.4.1	Cost	127
4.4.2	Execution Time	127
4.4.3	Memory and Power Consumption	128
4.5	Conclusion	128

4.1 Introduction

In this chapter, we used several different methods to evaluate the software solution performance in order to precisely assess particular properties of the solution required for its straightforward functioning.

Execution time is a fundamental criterion of the performance of transportation management systems, including PSEV and DISV, considering that delays are likely to result in serious disruptions. The smooth functioning can only be ensured if all messages are added to the smart contract promptly because the mining process depends on solving exceptionally complex problems. The proposed prototype is a real time application, which ensures instant communication and processing of payment between vehicles and other actors in the transportation system. The next criteria to be assessed is Power and Memory consumption because typical IoT devices have low power and computational capacities.

In addition, security of transactions is another criterion to be tested when Blockchain technology is used. In particular, it is critical to pay attention to the implementation methods in the applications. OWAPS guidelines for mobile and web applications is used for ensuring the security of the Blockchain-based solution. Meeting these criteria ensures an adequate validation and verification approach for the Blockchain-based applications.

Furthermore, there are additional properties of the developed solution that enhance its performance and ensure flawless operations, namely immutability, integrity, availability, and costs. This chapter evaluates these performances in order to assess the solution's overall efficiency.

4.2 Blockchain IoV Solution for Vehicles communication (DISV)

The performance of software solution can be assessed by applying different methodologies [270,271]. In particular, it is fundamental to evaluate the specific properties needed for the smooth functioning of the solution. The main properties of the proposed solution are execution time, costs, availability, integrity, immutability, and security as shown in Figure 4.1. All these properties must provide the highest performance in order to allow the flawless operations of the solution. Thus, this work will address these properties to assess the overall performance of the solution.

4.2. BLOCKCHAIN IOV SOLUTION FOR VEHICLES COMMUNICATION (DISV)

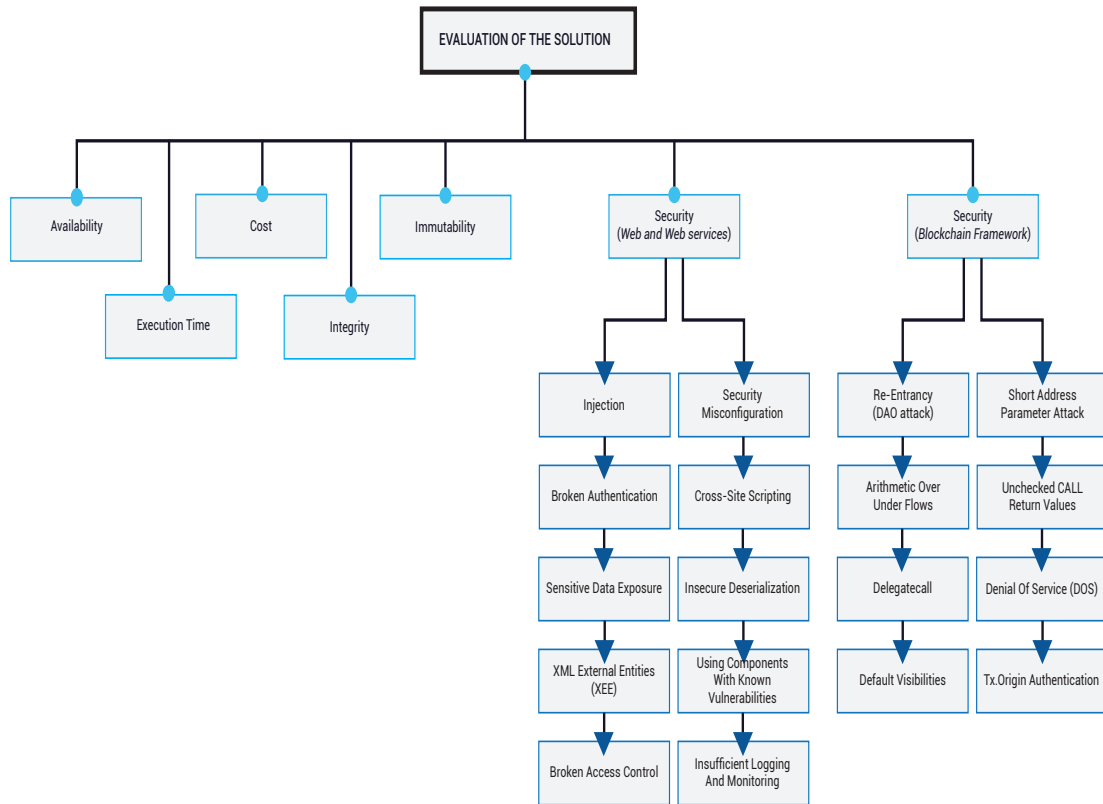


Figure 4.1: System evaluation diagram.

4.2.1 Costs

The Testnet of the Ethereum network was used to deploy the prototype of the smart contract. In this section the costs of the creation and execution of the smart contract are analyzed. The following values, valid in January 2020 were used: 1 gas_1 wei (0.000000001 ETH) and 1 ETH \$161,92 US. The minimum gas value to be used in a transaction was set at 1 wei; the average gas value was approximately 0.006845 Ethereum at the moment of analysis.

$$1 \text{ Gas} = 0,006845 \text{ Ethereum (ETH)}$$

$$\text{Gas Price} = 6,138,887 \text{ Gwei}$$

Table S1 presents the execution costs of various functions of the app. As summarized, the creation and deployment of the prototype on the Blockchain is the most expensive, \$0.07572 US. However, it

4.2. BLOCKCHAIN IOV SOLUTION FOR VEHICLES COMMUNICATION (DISV)

is worth mentioning that this is a one-time cost to set up and initialize the system. Moreover, this cost can be minimized by removing Truffle, which is the framework used for building, deploying, and managing smart contracts. It also includes features such as a “Migrations” contract to manage the deployment cycle.

Table 4.1: Costs of the different functions in the Smart Contract based on 1 ETH = 161,92 USD and 1 gas = 0,000000001 ETH Rates.

Function	Gas Used	Price
Deploy Contract	389,473	0.07572 (one time / Truffle)
SendMessage	140,345–257,488	0.02486–0.0456
Get Message	0	0

The “Set Message” function is not expensive, as it costs on average \$0.03523 US. However, there are significant variations of the costs of “Set Message” functions due to variable message input lengths. Nevertheless, the cost of one byte is set at 136 gas (\$0.00003 US). In contrary, the “Get Message” function does not require any additional cost because mining is not necessary while receiving messages from the blocks and no updates are needed for the smart contract.

4.2.2 Execution Time

One of the main criteria for evaluation of transportation management systems such as DISV is execution time. In fact, even a minor delay in sending or receiving messages can lead to severe disruptions of the system. For the proper functioning of the framework for communication between vehicles and all other actors in the transportation system based on Blockchain technology, it is fundamental to ensure the timely addition of each message to the smart contract as the mining process relies on complex problem solving.

Considering that the proposed prototype is a real time application, execution time is very important. In computational tests, the call times for each function of the Android application are measured. In order to assess the performance of the Ethereum private Blockchain proposed solution, a server with a Configuration of 64 GB Ram and a Core i7-000 was used. Figure S6 shows that the response time of the server of the GetMessage function is significantly shorter than the response time of the SendMessage. In fact, calling the GetMessage function needs between 1 milliseconds and up to almost

4.2. BLOCKCHAIN IOV SOLUTION FOR VEHICLES COMMUNICATION (DISV)

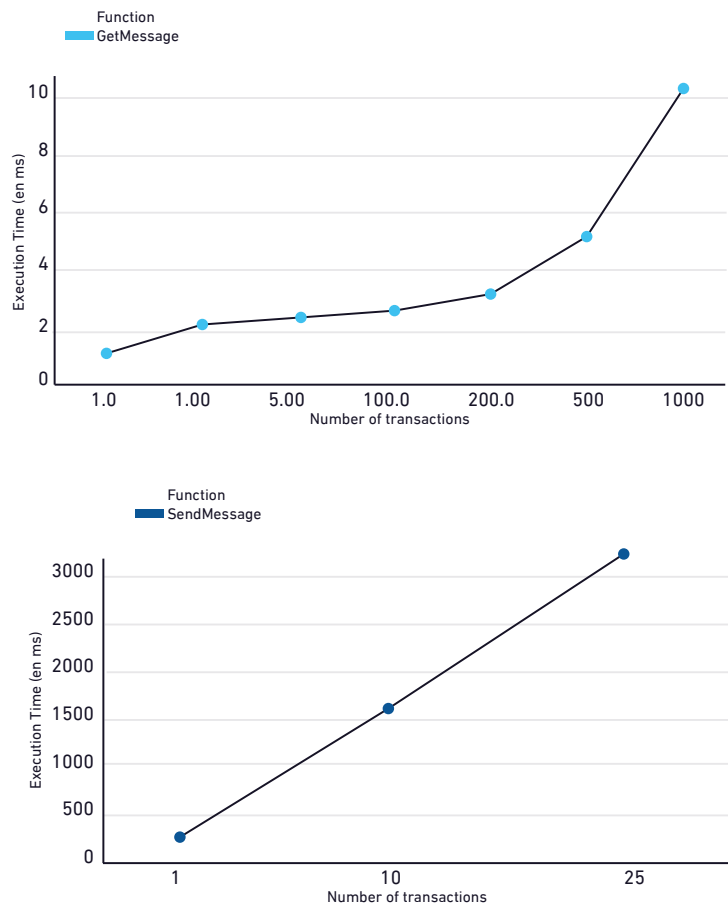


Figure 4.2: Execution time of the different functions in the Decentralized IoT solution for Vehicles communication (DISV) in milliseconds.

4.2. BLOCKCHAIN IOV SOLUTION FOR VEHICLES COMMUNICATION (DISV)

10 milliseconds when the server get 1000 requests. Therefore, there is no big difference in execution time between calling the GetMessage function one time or hundreds of times. However, calling the SetMessage function requires a lot of time because the message should be mined in order to add it into the smart contract. When the server receives ten requests of calling the SetMessage function, it takes 1.64 s and more than 90 s in the case of five hundred requests. Due to several factors, the computational study showed that is not recommended that the of list SetMessage exceeds 25 messages. Real time execution is one major property of DISV since any transaction should take a very small running time. Therefore, it was recommended in the proposed architecture to use the Blockchain layer in every zone so the server receives and sends a small number of messages. In addition to using many Blockchain layers, the Android application removes the old and duplicated messages so the smart contract contains only the necessary messages. Using the recommended architecture, in DISV, the response time of the messaging server is usually between 0 and 3 s. Hence, DISV can be considered a real-time application (RTA).

4.2.3 Memory and Power Consumption

Since DISV uses Blockchain for the IoT, it is critical to evaluate Power and Memory consumption, considering that IoT devices usually have very low power and computational capacities. A Huawei P8 Lite with a Configuration of 2 GB Ram, Li-Po 2500 mAh battery and a Hisilicon Kirin 620 Processor was used in computation tests in the demo. Figure S7 demonstrates that the memory consumption of the developed Android solution is significantly lower than the consumption of other commercial applications, such us Facebook app (134 MB), WhatsApp (106 MB), and Skype (233 MB). Regarding Energy Consumption, the proposed solution consumes electricity as average 23.43 mAh which is similar to Skype and Facebook mobile applications, which consume respectively 21.66 mAh and 18.56 mAh, as illustrated in Figure S8.

4.2. BLOCKCHAIN IOV SOLUTION FOR VEHICLES COMMUNICATION (DISV)

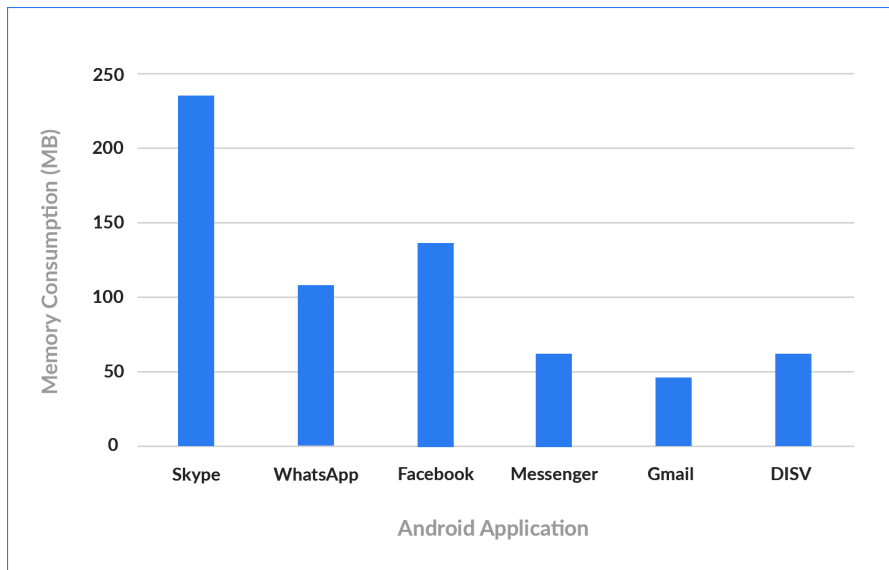


Figure 4.3: Comparing Memory Consumption of DISV with commercial mobile applications.

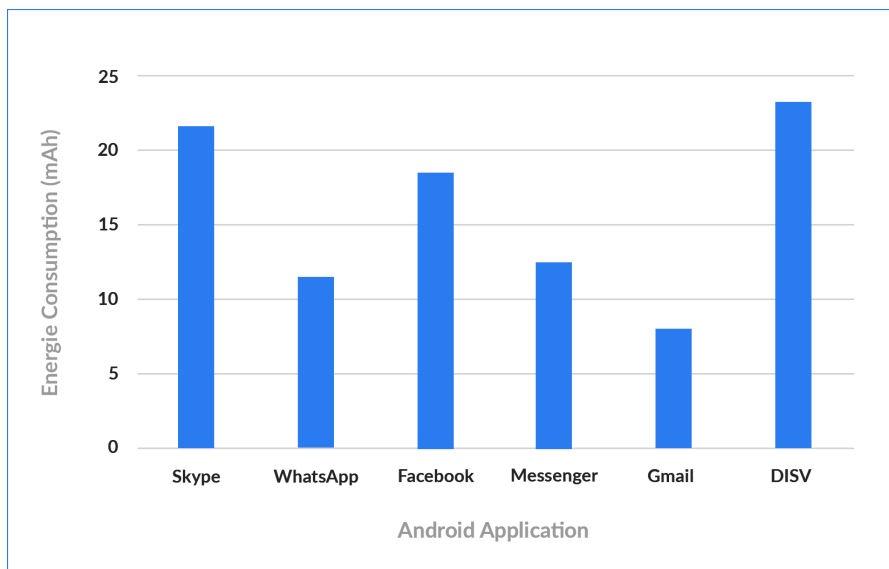


Figure 4.4: Comparing Energy Consumption of DISV with commercial mobile applications.

4.2.4 Availability

Another critical property of transportation management systems such as DISV is availability. More precisely, even the minor temporary shutdown of the system is likely to lead to traffic congestion and crashes. Availability means that a system is online and ready for access at any time. A variety of

4.2. BLOCKCHAIN IOV SOLUTION FOR VEHICLES COMMUNICATION (DISV)

factors can cause a shutdown of the system (off-line), ranging from planned downtime for maintenance to sudden failure. The decentralized and robust nature of the Blockchain prevents attacks such as a Denial-Of-Service (DoS) attack [272,273], which only target nodes [274], as the central party cannot be a single point of failure [273,275]. Yet, the distribution of the Blockchain is not complete. The mining power is typically limited to miners residing in approximately the same location. Thus, this enables isolating them by hijacking some border gateway protocol (BGP) prefixed with a routing attack employing the internet infrastructure [274]. Considering the comprehensiveness of the Internet, the Ethereum Blockchain network must be always reachable. Solutions that are centralized but whose databases are distributed can be vulnerable to routing attacks because of potentially hindered communication with and between the physical databases. Owing to the resiliency of the Blockchain malicious and damaged nodes on the network can be handled [276,277].

One of the greatest threats to the availability of a Blockchain solution is 51% attacks, which is the ability of someone controlling a majority of network hash rate to revise transaction history and prevent new transactions from confirming. As opposed to a public chain, these messages delays are combined with the heterogeneous power of miners in such a private chain could easily allow a 51% attack and lead to the Blockchain anomaly.

4.2.5 Integrity

Integrity is another fundamental property of the systems which exchange sensitive data among the users. Thus, it is necessary to assess the data integrity property of the proposed software solution. Data integrity refers to the accuracy and reliability of data through the whole life cycle. It is critically related to the concept of data security and it remains unchanged in regard to its complete state. It is essential for ensuring security to keep data consistent throughout its life cycle. The reliability of data refers to compliance with the following standards:

- The accuracy of data – free from errors and confirmed by the protocol.
- The originality of data – accessible sources and preservation in the original form.
- Contemporary – data must be recorded at the exact time it was executed and observed.

4.2. BLOCKCHAIN IOV SOLUTION FOR VEHICLES COMMUNICATION (DISV)

- Legible – easy to understand, record permanently, and preserve original entries.
- An attributable – clear demonstration of who observed and recorded data, at what time, and what it is about.

Cryptographic Hashing and Merkle Trees are in charge of keeping the integrity of the data on public and private Blockchain intact. There are three main advantages of Merkle trees [277]. First, they ensure the validity and integrity of data. Second, their proofs are fast and computationally easy requiring less disk space or memory. Third, their management and proof needs minimal information to be transmitted across networks. Moreover, Cryptographic Hashing is critical for keeping the integrity and security of data recorded on Blockchain. Encryption guarantees security, whereas integrity is achieved by ensuring that signatures update when the data is changed. Therefore, considering that the DISV is based on Blockchain technology, it ensures the communication between vehicles which maintains data integrity at all times.

4.2.6 Consistency

One of the major criteria for a new system evaluation is consistency which refers to the requirement that a series of measurements of the same project yields comparable results when different raters perform it by the same method. The proposed solution has employed the Ethereum Blockchain to build the consensus mechanism. Accordingly, as explained in [277], explicit reconciliation processes are not required. The consistency mechanism is based on the assumption that the branch behind the most Proof-of-Work represents the real branch. To ensure consistency, each block in the Blockchain accepted by a node preserves the consistency of the local replica of the database. In a case of a temporary disagreement among the nodes on the real consistent truths, Proof-of-Work enables the automatic resolving of the fork. Honest nodes cannot under any circumstances adapt to inconsistent chains. Within the network, the deeper buried blocks in the chain are always consistent. Considering Proof-of-Work prevents unsolvable reconciliation process, it is evident that the Blockchain ensures consistency in the proposed DISV system.

4.2.7 Confidentiality

Confidentiality in the context of computer systems allows authorized users to access sensitive and protected data. Thus, it is necessary to build in specific mechanisms to ensure confidentiality and safeguard data from harmful intruders. Confidentiality in a Blockchain setting allows involved parties to perform a transaction while preventing other participants from finding out certain facts or details about the transaction. Unlike public Blockchain such as Ethereum and Bitcoin that reject the concept of confidentiality and all their transactions are in the clear, confidentiality in private Blockchain can be handled if both the transaction and the identities of participating nodes are protected. To achieve these requirements, the proposed solution satisfies the following requirements:

- An unauthorized third party must be able to identify the counterparties to a transaction in a Blockchain until the counter parties reveal that information.
- Transaction details must be invisible to the person who is not involved in that particular transaction until the participating parties don't disclose their information.

4.2.8 Immutability

Immutability implies that alteration is not possible after the creation. To modify a transaction from history, it is necessary to re-mine all the blocks before the given block, which will reflect subsequently in each copy of the ledger in the network. It would also require rebuilding the Merkle tree of the block in which the transaction is located and redoing of all the proof of work for that block. Moreover, as the next block stores the hash of this block, it must also be re-mined. The subsequent block must be edited with the new "previous block hash", which leads to a different block hash. Such a hash in certain cases would not match the set difficulty level, which implies re-mining of the block. In fact, the re-mining will have to occur until the final block in the chain. Simultaneously, while the miner re-mines old blocks, new blocks will be added to the chain. It means that in addition to re-mining the previous blocks to edit a historical record, the miner will have to edit newly generated blocks at the same time. This action is practically impossible due to the enormous computing power required.

4.2. BLOCKCHAIN IOV SOLUTION FOR VEHICLES COMMUNICATION (DISV)

Accordingly, immutability is guaranteed in the proposed DISV system.

4.2.9 Security

Open Web Application Security Project (OWASP) Foundation's [278] top web vulnerabilities was employed to assess the security of websites, web services, and the central server of the DISV system. OWASP is a non-profit organization whose mission is to offer practical and unbiased information on application security to security professionals and developers. It primarily focuses on the critical vulnerabilities of web apps. The vulnerability assessment analysis detected ten of the most common attacks presented in Table 4.2 together with security requirements needed to minimize vulnerability combat attacks. The recommended requirement is incorporated into the system. However, newly and complex attacks occur on a daily basis. Thus, applying the recommended prevention cannot entirely prevent the attacks, but rather minimize the possibility of damaging or hacking the system. The architecture and technology of Ethereum in regard to the smart contract have certain vulnerabilities, which are listed in Table 4.3.

Attacks on Ethereum are most commonly in relation to the external call. However, some attacks rely on particular functions and employ a loop in the smart contract.

The security of the proposed solution is considered high because it is using private Blockchain. In Table 4.3, several recommendations are presented.

4.2. BLOCKCHAIN IOV SOLUTION FOR VEHICLES COMMUNICATION (DISV)

Table 4.2: Open Web Application Security Project (OWASP) top web vulnerabilities and security and privacy requirements.

No.	System Attack	Description	Security and Privacy Requirements
A1	Injection	Injection Attacks (CRLF, LDAP, and SQL injection) take part when untrusted data is sent by an attacker to an interpreter and then executed as a command without adequate authorization.	The prevention is done by the system by ensuring that data is separate from queries and commands.
A2	Broken Authentication	Broken Authentication and Session Management vulnerabilities refer to users being able to work around sessions and authentic mechanisms or manipulate them.	The prevention is done by the system through strong storage mechanisms and password policies.
A3	Sensitive Data Exposure	Applications and APIs which do not have proper protection of sensitive data such as username and passwords and financial information. Accordingly, attackers can steal identities and commit fraud upon accessing such information.	It is necessary to have SSL incorporated into the system and to transfer sensitive data only with encryption such as AES-256. The detection of insecure obfuscation techniques is needed.
A4	XML External Entities (XEE)	Exploitation of vulnerable XML processors by attackers by uploading XL and inserting hostile content in an ML document. It can also be done by exploiting vulnerable integrations, dependencies, and code.	It is recommended to incorporate the Rest paradigm into the system and use data formats such as JSON. Also, avoiding the serialization of sensitive data is needed.
A5	Broken Access Control	This vulnerability occurs when users can access certain applications functionalities that are not intended for their use. Accordingly, they can modify a URL as a way to reach other functionalities.	It is necessary to incorporate a strong access control mechanism into the system.
A6	Security Misconfiguration	This is the case of insecure and outdated configurations and also not adequate protection of directories and files by a web server.	All components of an application, including an operating system, language runtime, and server must be suitably hardened following recommended best practices.
A7	Cross-Site Scripting	Attackers perform scripts using XSS in the victim's web service endpoint or browser to redirect the user to malicious sites, deface websites or hijack user sessions.	To prevent malicious data from harming the database or website the system must render the correct data (Validate the Data).
A8	Insecure Deserialization	Insecure deserialization flaws allow an attacker to execute remote code in the application, elevate privileges, carry out injection and delete or tamper with serialized (written to disk) objects.	The system must incorporate SSL.
A9	Using Components With Known Vulnerabilities	Vulnerable components – frameworks, libraries, etc must run with full privilege.	The system must use approved enterprise libraries.
A10	Insufficient Logging And Monitoring	The detecting time of a breach is typically measured in weeks and sometimes months. Thus, in sufficient logging and ineffective integration, relevant security incident response systems allow attackers to reach other systems and become a persistent threat.	Monitoring systems such as Appdynamics and Dynatrace have defined rules and send proactive alerts. They should be incorporated into the system.

4.3. BLOCKCHAIN IOV SOLUTION FOR PAYMENT (PSEV)

Table 4.3: Ethereum top web vulnerabilities and Security and privacy requirements.

No.	System Attack	Description	Security and Privacy Requirements
AE1	Re-Entrancy (DAO attack)	The smart contract of Ethereum can call and use codes of other external contacts, which can be hijacked and subsequently forced to conduct new codes through, for instance, a fallback function.	The transfer function should not send more than 2,300 gas with the external call, as it prevents the destination address/contract from calling another contract.
AE2	Arithmetic Over/Under Flows	The Ethereum Virtual Machine (EVM) determines fixed-size data types for integers. Attackers can exploit Variables in Solidity in a case of unchecked user input and performed calculations in numbers outside the range of the data type storing them them.	Protecting against under/overflow vulnerabilities is performed by designing or using mathematical libraries to replace the standard math operators, namely addition, subtraction and multiplication.
AE3	Delegatecall	Ethereum developers use the CALL and DELEGATECALL opcodes to modularize their code. However, DELEGATECALL can result in unintended code execution.	Solidity holds the library keyword to implement library contracts. (Check the Solidity Docs) Consequently, the library contract is non-self-destructable and stateless.
AE4	Default Visibilities	Solidity functions include visibility specifiers to dictate how functions are permitted to be called. Incorrect use of those specifiers results in serious vulnerabilities	The visibility of all functions should be specified in the contract.
AE5	Short Address/ Parameter Attack	The parameters passed to the smart contract are encoded by the ABI specification. Sending encoded parameters shorter than the expected parameter length is possible.	Prior to sending the inputs to the Blockchain, they should be validated by the system.
AE6	Unchecked CALL Return Values	Performing external calls in solidity can be done in several ways. Typically, the transfer method is used to send ETH to external accounts; while the send () function is employed for versatile external calls. Moreover, the CALL is used directly in solidity.	In all possible cases the transfer() function should be used instead of send() transfer() reverts when the external transaction reverts.
AE7	Denial Of Service (DOS)	A DDoS attack on Ethereum Blockchain indicates that an attacker intends to use all resources of the network so that minors cannot record or cater to other transactions.	Contracts must not loop through data structures which allow artificial manipulation by external users.
AE8	Tx.Origin Authentication	Contracts authorizing users by the tx.origin variable are vulnerable to phishing attacks. These attacks trick users to carry out authenticated actions on the vulnerable contract.	Do not use tx.origin for authorization in smart contracts.

4.3 Blockchain IoV Solution for payment (PSEV)

Different methods can be employed to evaluate the performance of the software solution [279] [280]. The most critical part of this system is the assessment of the particular features required to allow the solution to function properly. The principal features of the proposed solution are integrity, cost, consistency, confidentiality, execution time, immutability, and memory and power consumption. The smooth operation of the solution requires that all these features achieve high performance. Therefore,

4.3. BLOCKCHAIN IOV SOLUTION FOR PAYMENT (PSEV)

this work evaluates specific features to evaluate the general performance of the proposed solution.

4.3.1 Cost

In standard Blockchain deployment, cryptocurrencies are incentives employed to reward network nodes/operators, which ensure consensus and data integrity, and accordingly, ensure the decentralized ecosystem. However, considering costs required for storing information and performing computations, Blockchain users must pay for incentives. Therefore, this work examines the costs related to the services run by a DApp with these employments. Furthermore, the study compares those expenses with the costs incurred in widely used centralized and proprietary systems. In particular, it is necessary to perform cost estimation when a DApp provides services to numerous providers and clients. Moreover, the assessment of the cost-effectiveness is critical when Blockchain is used to replace conventional elements of the parking systems in order to improve interoperability. Testnet of the Ethereum network have been used for the deployment of the prototype of the smart contract and accordingly, to assess how cost-effective is the solution. This part of the work evaluates the expenses needed to create and execute the smart contract. The following rates, valid in May 2020 were used: 1 gas 1 wei (0.000000001 ETH) and 1 ETH 209 US. The lowest gas value that can be used in a transaction is 1 wei; the average gas value was approximately 0.006252 Ethereum at the time of evaluation.

$$1 \text{ Gas} = 0.006252 \text{ Ethereum (ETH)}$$

$$\text{Gas Price} = 4,652,309.7157 \text{ Gwei}$$

Table 4.4 gives an overview the execution costs of the functions that are most commonly called. As summarized, the highest cost of \$30.7 US incurs to generate and deploy the prototype on the Blockchain. However, it is worth mentioning that only one payment is required to establish and deploy the system. Moreover, if Truffle is removed, this cost decreases. Truffle is the framework used to build, deploy, and manage smart contracts. It possesses a Migrations contract for managing the cycle of deployment. Our analysis revealed that “Add Vehicles”, “Send Message”, “Buy Ticket”, and “Sell Ticket” are the most frequently called functions. The “Buy Ticket” costs on average \$0.028 US. Furthermore, mining is not required to get messages from the blocks when calling the functions “Get Vehicles By Address”, “Get All Vehicles”, “Get Parking By Address”, “Read Message”, and “Get

4.3. BLOCKCHAIN IOV SOLUTION FOR PAYMENT (PSEV)

Table 4.4: Functions cost of the Smart Contract based on the used Rates.

System	Function	Gas Used	Price (US ~ Dollars)
	Deploy System	0,14437926 ETH	\$30,70
	Add Vehicles	132797	\$0.08446
Access Management	Get Vehicles By Address	0	0
	Get All Vehicles	0	0
	Add Parking	150346	\$0.09561
	Get Parking By Address	0	0
	Get All Parking	0	0
	Communication Management	Send Message	89280
	Read Message	0	0
Parking Payment Management	Sell Ticket	166854	\$0.10613
	Get Number Of Tickets	0	0
	Get Tickets For Sale	0	0
	Buy Ticket	43078	\$0.2739

Tickets For Sale”. Consequently, updates are not needed for the smart contract, and these functions do not incur further costs.

4.3.2 Execution Time

The system performance tests were carried out in PSEV. The tests were performed on a server with the following configuration: 8 GB Ram and a Core i7-000. Table 4.5 demonstrates that the server’s response time of the functions “GetTicketsForSale”, “ReadMessage”, “GetParkingByAddress”, “GetAllVehicles”, and “GetVehiclesByAddress” are significantly shorter than the response times of other functions as mining is not required for their interaction with the smart contract. Nevertheless, it takes minimum 2 and maximum 157 milliseconds to call any functions in the smart contract. Accordingly, the scalability of PSEV across large populations of users has been proven. The execution time is a fundamental criterion for the assessment of communication systems such as PSEV-Communication. More precisely, minor delays in sending or receiving messages would result in serious disruptions of the system. Thus, adding instantly all messages to the smart contract is critical to ensure the adequate operation of the framework for communication among all participants in the transportation system. It is worth mentioning that the mining process is based on complicated problem solving.

Figure 4.5 demonstrates that the function server takes significantly less time to respond than

4.3. BLOCKCHAIN IOV SOLUTION FOR PAYMENT (PSEV)

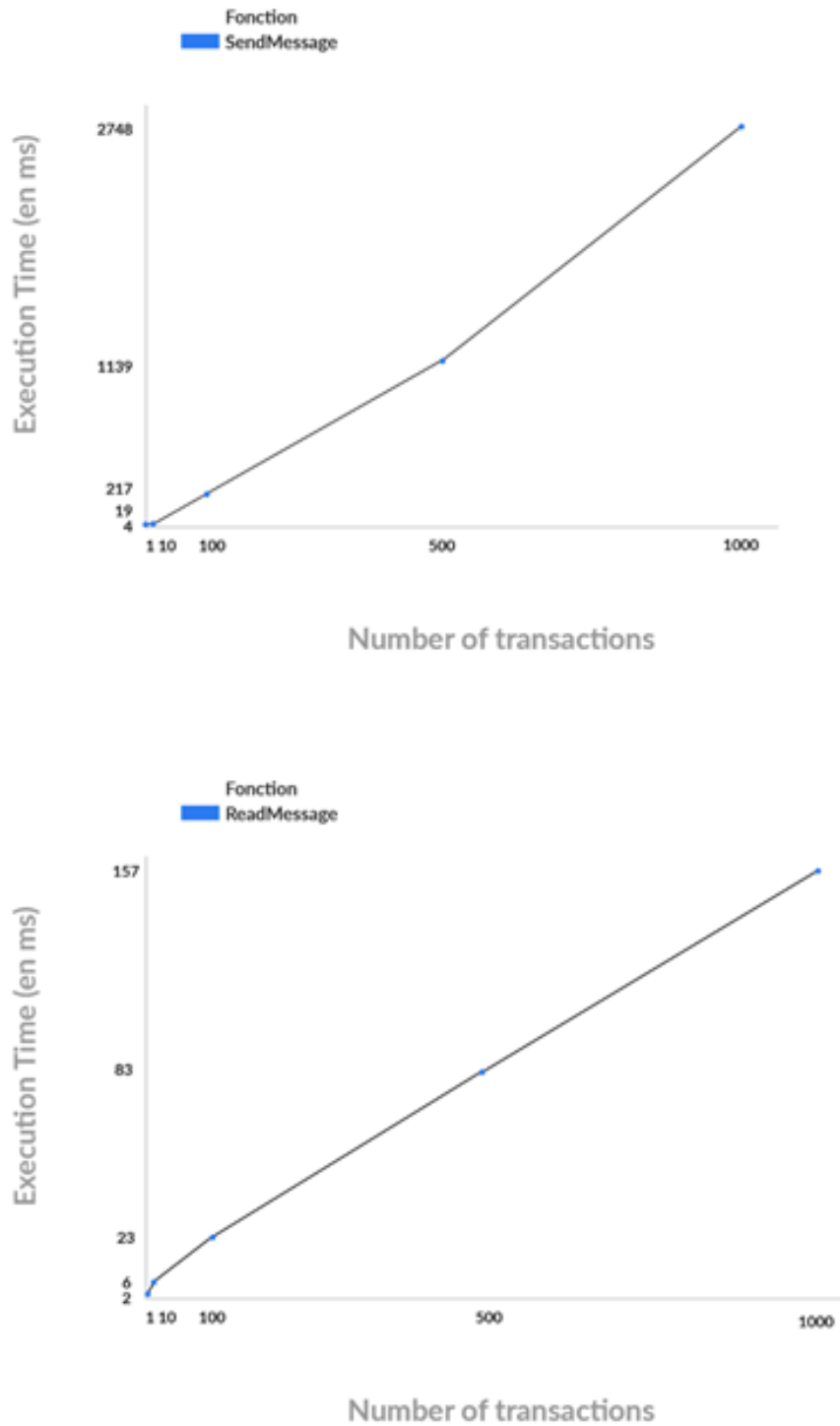


Figure 4.5: Execution time of most called functions of PSEV-Communication.

4.3. BLOCKCHAIN IOV SOLUTION FOR PAYMENT (PSEV)

Table 4.5: Execution time of most called functions of the Smart Contract in milliseconds.

System	Function	Time (ms)
Access Management	Add Vehicles	28
	Get Vehicles By Address	7
	Get All Vehicles	11
	Add Parking	32
	Get Parking By Address	9
	Get All Parking	15
Communication Management	Send Message	4
	Read Message	2
Parking Payment Management	Sell Ticket	43
	Get Number Of Tickets	18
	Get Tickets For Sale	12
	Buy Ticket	57

the More precisely, it takes between 2 milliseconds and up to almost 157 milliseconds in the case of 1,000 requests per server to call the function. Accordingly, it does not make a significant difference in execution time if the function is called once or even 100 times. On the other hand, calling the function requires much more time as it is necessary to mine the message for adding it into the smart contract. 19 milliseconds are needed when ten requests of calling the “Send Message” function are received by the server and more than 2.748 seconds in the case of 1,000 requests in PSEV, as the messaging server response time typically takes between 0 and 3 seconds. Therefore, PSEV has been proven to be a real-time application. Moreover, the performance tests indicated, when compared with other similar solutions [281], the proposed application is faster and more scalable. More precisely, the proposed application send up to 1145 message in 3 seconds, whereas similar solutions can send only 25 to all the participants in intelligent transportation systems.

4.3.3 Integrity

For the systems that share sensitive data among the participants, it is critical to ensure integrity, as it is their fundamental feature. Accordingly, the assessment of the data integrity feature of the tested solution is crucial. Data integrity refers to how reliable and accurate is the data during the life cycle. Data security and integrity are highly interconnected. For data to be uncorrupted, it has to be whole and unchanged concerning its complete state. Thus, in order to ensure security, data must be consistent during the entire life cycle. The following standards must be complied with for ensuring

4.3. BLOCKCHAIN IOV SOLUTION FOR PAYMENT (PSEV)

the data reliability:

- The accuracy of data – data do not contain errors, and they are verified by the protocol.
- The data originality – sources can be accessed, and data preserve their initial form.
- Contemporary – data are being recorded at the moment of execution and observation.
- Legible – easy understanding, permanent recording, and preservation of original entries.
- It can be monitored who saw and recorded data, at precisely what time, and the type of data.

The role of cryptographic Hashing & Merkle Trees is to maintain the integrity of the data on both public and private Blockchain unchanged. Merkle Trees [277] have the following main advantages. First, they have sufficient capacity to guarantee that the data are valid and with preserved integrity. Moreover, Merkle Trees require less disk space or memory because their proofs are prompt and easy to compute. Furthermore, to transmit the proofs across the networks and to manage them, minimal information is needed. In addition, Cryptographic Hashing keeps the data security and integrity recorded on Blockchain efficiently. Security is guaranteed through encryption, while signatures are updated every time when the data is modified, which ensures integrity. As the PSEV relies on Blockchain technology, data integrity is constantly ensured during the communication and payment operations.

4.3.4 Consistency

Consistency is a critical criterion for the assessment of the new system. Consistency refers to assessments of the same project that generate comparable results when various raters use the identical method. The developed solution uses the Ethereum Blockchain to establish the consensus mechanism. Therefore, following [277], the reconciliation process is not required. The assumption behind the consistency mechanism is that the branch of most of the Proof-of-Work is the real one. To ensure consistency, a node must accept each block in the Blockchain and the local copy of the database must be uncompromised. The automatic resolution of the work occurs the nodes are in the temporary disagreement on the real consistent truths. It is impossible that honest nodes adapt to inconsistent chains in any case. Thus, the blocks located deeper in the chain within the network, are consistent

4.3. BLOCKCHAIN IOV SOLUTION FOR PAYMENT (PSEV)

all the time. As Proof-of-Work prevents the unsolvable reconciliation process, the proposed PSEV system ensured consistency due to the Blockchain.

4.3.5 Confidentiality

Concerning computer systems, confidentiality concerning to the fact that only authorized users can access sensitive and protected data. Accordingly, specific mechanisms must be built to guarantee confidentiality and protect data from malicious attackers. In the context of Blockchain, confidentiality indicates that involved users can carry out a transaction, whereas other users cannot access specific information about the transaction. In the public Blockchain network, the notion of confidentiality is rejected, and accordingly, all the transactions are transparent. In contrast, private Blockchain ensures confidentiality to protect both the information on the transactions and the identities of participating nodes. The solution must satisfy the following requirements to achieve confidentiality:

- A person not participating in a given transaction cannot access the transaction details unless the participating parties share information.
- An unauthorized third party cannot access the identities of the participants of transaction in a Blockchain until they share them.

4.3.6 Immutability

Immutability prevents alteration after the creation. It would be needed to re-mine the blocks prior to the particular block to alter a transaction from history. Only in this case it could be rejected in all ledger copies in the network. Also, it is necessary to reconstruct the Merkle tree of the block holding the transaction as well as the proof of work. Considering that the upcoming keeps the hash of this block that one would have to be re-mined as well. The upcoming block would need to be modified with to include “previous block hash”. Accordingly, a new block hash would be created. Re-mining of the block is required if the hash cannot meet the proper difficulty level. More precisely, the re-mining must be continued until the last block in the chain is re-mined. However, new blocks are added to the chain in the same time when old blocks are re-mined and new simultaneously created blocks have to be re-mined as well. This action cannot be completed because of the needed computing power. Therefore, the proposed PSEV system ensures immutability.

4.3.7 Memory and Power Consumption

As PSEV uses Blockchain for the IoT, the assessment of memory and power consumption is required, taking into account considering low computational capacities and power of IoT devices. In the demo, a Huawei P8 Lite (2 GB Ram, a Hisilicon Kirin 620 Processor, Li-Po 2500 mAh battery) was utilized to perform the computational assessment. The results presented in Figure 4.6 indicate the

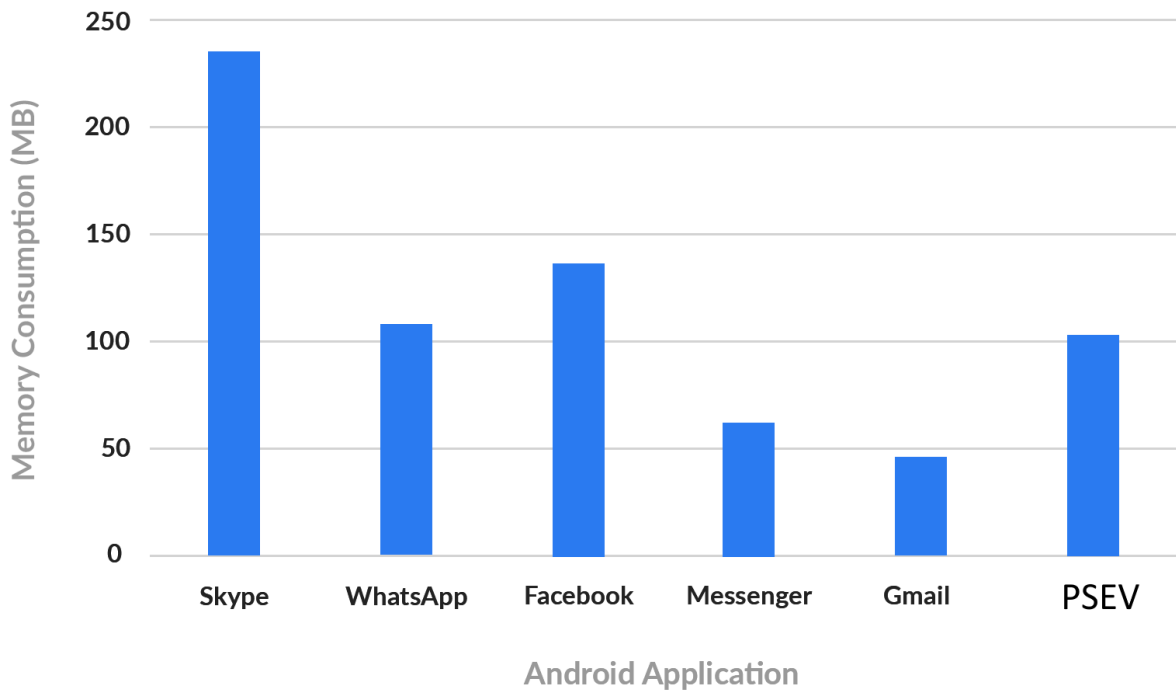


Figure 4.6: Comparison of memory consumption of PSEV and mobile applications commercially available.

proposed Android solution has lower memory consumption than numerous commercial applications, such as Skype (233 MB), Whats-App (106 MB), and Facebook apps (134 MB). Concerning energy consumption, on average, the developed solution consumes power as 25.43 mAh, which is comparable to the Facebook app and Skype app, using 18.56 mAh and 21.66 mAh, respectively, as shown in Figure 4.7.

4.3. BLOCKCHAIN IOV SOLUTION FOR PAYMENT (PSEV)

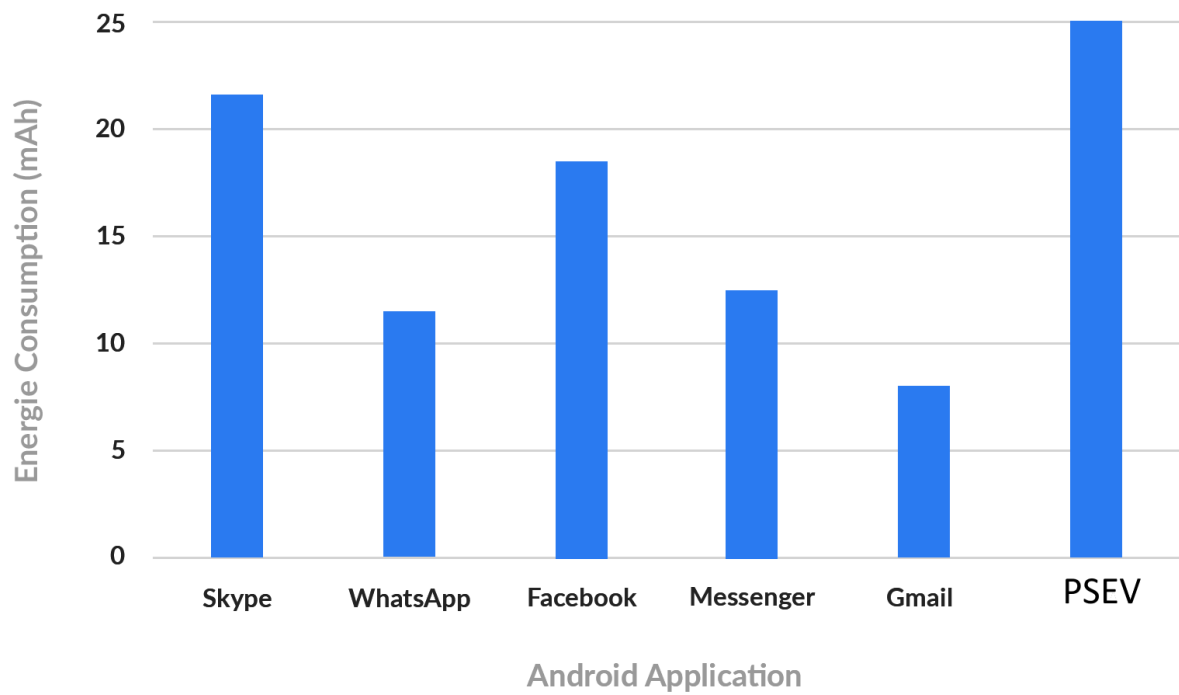


Figure 4.7: Comparing Energy Consumption of PSEV with commercial mobile applications.

4.3.8 Security

Some testing needs must be considered to ensure a secure transaction using Blockchain technology. Therefore, OWASP guidelines for mobile and web applications, Mythx, and SCSVS are applied to overcome challenges and ensure the security of Blockchain-based solution. Importantly, the testing depends on the methods of the implementation in the applications. These factors provide adequate proper validation and verification approach for the Blockchain-based applications:

4.3.8.1 Mythril and MythX

We conducted security tests to ensure the resiliency of the system against malicious attacks. The first test was performed to detect security vulnerabilities in our smart contracts, which represent the core of our system. In this test, we scanned the smart contracts against possible vulnerabilities using the security tool named MythX [282] a Smart contract security service for Ethereum. MythX is divided into three tools: Maru for static analysis of smart contracts, Harvey for smart contract fuzzing, and Mythril for EVM bytecode symbolic execution. This service is aimed at analyzing the security of smart contracts at compiled bytecode and source code levels. This test did not reveal any vulnerabilities in the smart contract listed in the MythX Smart Contract Weakness Classification (SWC) Registry. For instance, the MythX report did not register assert violation, unprotected Ether withdrawal, or integer overflow/underflow. However, although smart contracts do not contain vulnerabilities as they are based on the Blockchain, it should be underlined that the system is prone to 51 percent attacks.

4.3.8.2 OWASP

OWASP is a non-profit organization that provides unbiased, practical guidelines to developers and security professionals by identifying primarily crucial web apps vulnerabilities. Security Open Web Application Security Project (OWASP) Foundation's [278] top web vulnerabilities were used for the assessment of the security of the central server of the PSEV system, web services, and websites. Table 4.6 shows ten most common attacks identified by the vulnerability assessment analysis and necessary security requirements that should be incorporated into the system to reduce vulnerability combat attacks. However, it is noteworthy mentioning that the attackers continually perform increasingly complex attacks. Therefore, it is not possible to completely eliminate the attacks. Instead, the

4.3. BLOCKCHAIN IOV SOLUTION FOR PAYMENT (PSEV)

prevention aims to minimize the damage and decrease chances for hacking the system. Table 4.7 outlines specific vulnerabilities of Ethereum's architecture and technology regarding smart contracts. The most common attacks on Ethereum are related to the external call; however, the attackers also use a loop in the smart contract by exploiting specific functions. The proposed solution is based on private Blockchain, and accordingly, it is of high security.

4.3.8.3 SCSVS

Smart Contract Security Verification Standard (v1.1) [283] represents a list aimed at standardizing the smart contract security for vendors, security reviewers, architects, and developers. The purpose of this checklist is to minimize vulnerabilities and security problems. Therefore, it offers guidance for all stages of the smart contract development cycle, starting with design and ending with implementation. The primary objective is to establish a high quality code of the smart contracts. Accordingly, it is required to identify and mitigate vulnerabilities. Furthermore, the objective of Smart Contract Security Verification Standard (v1.1) is to offer a straightforward and reliable assessment of the security of smart contracts regarding the percentage of SCSVS coverage and a checklist for security reviewers. In this work, the recommendation of Smart Contract Security Verification Standard was respected in the following Area : In the domain of architecture, design, and threat modelling, we tested the verification of a component that uses events to supervise the contract activity and of a policy for tracking new security bugs and updating the libraries to the newest secure version. Furthermore, we respected the consistency of the business logic in the contracts. Either none of the contracts should be allowed to conduct changes or all the contracts should be allowed. In addition, we applied the verification using code analysis tools to identify potentially malicious code. Moreover, we applied the newest version of Solidity and respected that the controlled, minimal acceptable value of cryptocurrencies is maintained on the contract.

In the domain of access control, we tested the presence of the principle of least privilege: other contracts must have particular authorization to access data or functions. Furthermore, we respected that calling external contracts is possible in urgent cases only. In addition, we respected the basis of the contract on the data given by the right sender. Thus, the contract must not be based on tx.origin value. Also, we applied the principle that access controls must maintain all user and data attributes in the trusted contract. Accordingly, the manipulation by other contracts is prevented if they do not

4.3. BLOCKCHAIN IOV SOLUTION FOR PAYMENT (PSEV)

Table 4.6: OWASP top web vulnerabilities and security and privacy requirements.

No.	System Attack	Description	Security and Privacy Requirements
A1	Injection	CRLF, LDAP, and SQL are injection attacks performed by sending an untrusted data to an interpreter. Consequently, the command is executed without proper authentication.	The system prevents such attacks by ensuring the separation between the data and queries and commands.
A2	Broken Authentication	The attacker uses Broken Authentication and Session Management vulnerabilities due to accessing the passwords' data base or the session ID. Broken Authentication can be defined as manipulation of authentication mechanisms.	Broken Authentication is prevented by selecting strong passwords and ensuring adequate storage mechanisms.
A3	Sensitive Data Exposure	The attackers can exploit Applications and APIs without protection of sensitive data, such as financial data, passwords, and usernames. As a result, the attackers can obtain and misuse data by for example committing fraud and stealing identities.	The system must entail SSL. AES-256 and similar encryptions should be used to transfer the data. All insecure obfuscation techniques must be detected.
A4	XML External Entities (XEE)	XL is uploaded by the attacker and hostile content is inserted in an ML document. Consequently, the attackers can manipulate vulnerable integrations, dependencies, and code and misuse vulnerable XML processors.	The system must entail the REST paradigm. The sensitive data serialization should be minimized. It is recommended to use JSON and similar formats.
A5	Broken Access Control	The attacker misuses specific functionalities of the app to exploit Broken Access Control. In addition, the attacker can modify an URL further to reach even more functionalities.	The system must entail a strong access control mechanism.
A6	Security Misconfiguration	If the web server fails to protect directories and files due to outdated and insecure configurations, the attacker can perform manipulation.	It is necessary to follow best practices for hardening properly all app components, including the operating system, language runtime, and server.
A7	Cross-Site Scripting	The victim's browser or service endpoint can be used to perform the script. The purpose of performing the script is to hijack the session or to redirect the user to malicious websites.	When adequate data validation is applied, the malicious data cannot attack the database or the website.
A8	Insecure Deserialization	The attackers use insecure deserialization flaws to delete or modify serialized objects written on the disk), perform the remote code in the application, upgrade privileges, or carry out injections.	The system must entail SSL.
A9	Using Components with Known Vulnerabilities	The full privilege is needed to run vulnerable components (e.g., frameworks and libraries)	The system should use exclusively approved enterprise libraries.
A10	Insufficient Logging and Monitoring	Breaches can be identified months after they were conducted if there are inadequate logging and ineffective integration and security incident response systems. Consequently, the attacker can access other systems and establish himself as a persistent threat.	The system must entail monitoring systems such as Appdynamics and Dynatrace to ensure that proactive alerts are sent by following predefined rules.

4.3. BLOCKCHAIN IOV SOLUTION FOR PAYMENT (PSEV)

Table 4.7: Ethereum top vulnerabilities and Security requirements.

No.	System Attack	Description	Security and Privacy Requirements
AE1	Re-Entrancy (DAO attack)	The smart contract of Ethereum is able to call and use codes of other external contacts. However, these contacts could be hijacked and consequently forced to perform new codes by a fallback function.	The transfer function must be prevented from transmitting more than 2,300 gas with the external call. As a result, the destination address/contract is not able to call another contract.
AE2	Arithmetic Over/Under Flows	Fixed-size data types for integers are determined by the Ethereum Virtual Machine (EVM). In the case of an unchecked user input, attackers are able to use variables and carry out calculations in numbers outside the range of the data type in which they are stored.	Mathematical libraries can be designed or used to replace multiplication, subtraction, and addition, which are standard math operations, to ensure protection against under/overflow vulnerabilities.
AE3	Delegatecall	Ethereum developers employ the DELEGATECALL and CALL opcodes for modularizing the code. Notably, DELEGATECALL may lead to unintended code execution.	Solidity possesses the library keyword for the implementation of the library contracts (See the Solidity Docs) As a result, the library contract is stateless and non-self-destructible.
AE4	Default Visibilities	Solidity functions entail visibility specifiers for determining in which cases functions can be called. If specifiers are used improperly, the attackers can exploit resulting vulnerabilities.	The contract must specify the visibility of all functions.
AE5	Short Address/Parameter Attack	The ABI specification encodes the parameters passed to the smart contract. It is possible to send encoded parameters shorter in length than the expected ones. /	The system should validate the input before it is sent to the Blockchain.
AE6	Unchecked CALL Return Values	There are several methods for carrying out external calls in solidity. Most commonly, ETH is sent to external accounts by using the transfer method. In the case of versatile external calls, the send () function is used. In solidity, the CALL is used directly.	In the case of the external transaction reverts, it is recommended to use the transfer() function instead of send() transfer() reverts whenever it is possible.
AE7	Denial Of Service (DOS)	A DDoS attack on Ethereum Blockchain implies the attacker employing resources of the network to prevent minors from recording or handling other transactions.	Contracts should be prevented from looping through data structures that permit artificial manipulation by external users.
AE8	Tx.Origin Authentication	Phishing attacks can exploit contracts in which tx.origin variable is used for user authorization. In these attacks, users are manipulated to perform authenticated actions on the vulnerable contract.	tx.origin should not be used for authorization in smart contracts.

4.3. BLOCKCHAIN IOV SOLUTION FOR PAYMENT (PSEV)

have particular authorization. This domain also verified the specification of the functions' visibility.

In the domain of Blockchain data, we tested if data stored in the contracts is not private or safe (private variables are also included). It is also necessary to respect and verify that the contract does not use employ literals as mapping keys for mappings, but that global constraints are employed for the prevention of the Homoglyph attack. Finally, we applied the identification of the list of sensitive data that the smart contract processed. In the domain of communications, we tested if the libraries were identified. The smart contract depends on the libraries for operation, but they are not a component of the app. We also respected that the contract does not employ hardcoded addresses, except in necessary cases. In the case of using a hardcoded address, the contract must be audited. In addition, we applied a centralized implementation of libraries and contracts that call external security services, that untrusted contracts and delegatecall are not used together, and that the result of low-level function calls such as call, delegatecall, and send are inspected and included as well. In the domain of gas usage limitations, we tested the anticipation, definition, and distinctive limitations that cannot be exceeded of the usage of gas in the smart contract. Therefore, gas exhaustion is not caused by malicious input and code structure. In this domain, we also respected that two kinds of the addresses are taking into account when employing the send function. It is more expensive to send to the contract address than to the personal address. In addition, we applied the principle that the iteration of the contract over unbound loops does not occur, the contract does not inspect if the address is a contract via extcodesize opcode, and that the contract does not create trivially pseudorandom numbers using the information extracted from Blockchain (e.g., seeding with the block number).

In the domain of business logic, we tested if the business logic flows of smart contracts follows a sequential step order that is pre-designed. Accordingly, none of the steps can be skipped. We also respected that the contract has business limits that are appropriately enforced, the contract logic is not based on the balance of contract (e.g., `balance == 0`), and the contract does not send funds automatically and ensuring that users perform separate transactions to withdraw funds. In addition, we applied that the inherited contracts do not include identical functions and careful specification of the order of inheritance.

4.4 Discussion

This section presents the comparison between the developed solution PSEV and DSEV. In this study, we compared the solutions in terms of costs, execution time, and memory and power consumption.

4.4.1 Cost

PSEV consists of three systems and each has particular functions. The Access Management System contains AddVehicles, GetVehiclesByAdress, AddParking, GetParkingByAddress, and GetAllParking functions. The Communication Management System includes SendMessage and ReadMessage functions. Finally, the Parking Payment Management System encompasses SellTicket, GetNumberOfTicket, and BuyTicket functions. The most expensive is the generation and deployment of the prototype on the Blockchain (\$30.7), followed by BuyTicket (\$0.2739), SellTicket (\$0.10613), AddParking (\$0.09561), AddVehicles (\$0.08446), and SendMessage (\$0.05677), while other functions do not incur costs. In contrast, DSEV has only three functions: DeployContract (\$0.07572 onetime), SetMessage (\$0,02486-0.0456), and GetMessage (0). Thus, the comparison revealed that deployment of the PSEV is more expensive because it contains more smart contracts and features.

4.4.2 Execution Time

The execution time is a fundamental criterion for the assessment of communication systems such as PSEV, as even minor delays cause serious disruptions. In PSEV, it takes between 2 and 57 milliseconds to call any function, which proves that it is a real-time application. In DISV, the messaging server's response time takes between 2 and 245 milliseconds. PSEV is also considered a real-time application. Importantly, PSEV is able to send up to 1145 messages in 3 seconds, which outperforms similar solutions such DISV that can send only 25 to all the participants in intelligent transportation systems. Furthermore, when comparing the two solutions It takes less time to send a message and get a message in PSEV but more time to sell and buy ticket.

4.4.3 Memory and Power Consumption

The assessment of memory and power consumption is required considering the low computational capacities and power of IoT devices. PSEV consumes power as 25.43 mA, which is similar to commercial applications such as Facebook, Skype, Gmail, Messenger, and WhatsApp. DSEV consumes electricity as average 23.43 mAh. More specifically, PSEV requires more than DSEV because it has more features. In conclusion, the comparison revealed that both solutions are real-time applications. Furthermore, PSEV has higher costs and memory and power consumptions. However, this is because PSEV is a more complex solution, with more functions to be executed.

4.5 Conclusion

In this chapter, we present the assessment of the features critical for achieving the high performance of the proposed solutions, namely cost, execution time, integrity, consistency, confidentiality, immutability, memory and power consumption. This assessment is aimed at evaluating if Blockchain can be used as a platform for IoV communications and parking payment securely and effectively. According to the results, the PSEV is a real-time application, faster and more scalable than current communication solutions. In addition, PSEV can resolve main challenges of V2X communications such as security and scalability. Finally, PSEV and DISV allow data sharing and cooperation among the participants in intelligent transportation systems, functioning of the Advanced Driver Assistance Systems (ADAS), and improve the overall security and safety of the transportation system.

Conclusion

Conclusion

“Vehicular Ad-hoc Networks” (VANETs) have been revolutionized and transformed into the “Internet of Vehicles” (IoV) owing to the innovative IoT technologies. The IoV uses information platforms, vehicle navigation systems, smart payment terminal devices and mobile communication technology to establish the real-time data interaction between vehicles and between pedestrians, vehicles, infrastructures. By being connected to the internet, vehicles that are part of the IoT become connected to traffic information systems and other vehicles as well. However, considering the exchange of sensitive data and high connectivity, the implementation can compromise privacy and security and accordingly, vehicles are prone to malicious attacks.

In this study, we developed two solutions using Blockchain technologies in the IOV to respond to related security and privacy challenges and avoid the possibility of the connected systems to become victims of cyberattacks and reveal sensitive information. Our first proposed solution is “DISV,” which stands for a Decentralized IoT Solution for Vehicles communication. DISV is based on the Blockchain and its aim is to improve the security of the communication on the IoV. DISV allows the participant of the IoV networks to receive messages whereas those messages are simultaneously added to the chosen Blockchain layer.

The second proposed solution is called “PSEV.” This solution is based on Blockchain and uses Ethereum to generate a smart payment system for parking (PSEV-Payment). While designing this solution, we followed the assumption that private parking space owners want to rent their parking lots when they do not use them. Moreover, this solution also operates in real-time and allows instant communication between all the participants in the transportation system. In addition, PSEV and DISV allow data sharing and cooperation among the participants in intelligent transportation systems, functioning of the Advanced Driver Assistance Systems (ADAS), and improve the overall security and safety of the transportation system.

In this study, we evaluated the most important properties of the solution, namely memory and power consumption, immutability, confidentiality, consistency, integrity, execution time, and cost. This evaluation was aimed at assessing if the platform based on the Blockchain can ensure effective and secure IoV communication and parking payment. According to the results, DISV and PSEV are proven to be not only real-time solutions, but they are more scalable and faster than current

communication solutions. Moreover, the study demonstrated that DISV and PSEV can contribute to the solving most critical challenges of vehicle-to-everything (V2X) communication by enhancing scalability and security.

Perspectives

The Blockchain-based solutions can significantly improve the ITS and more particularly by enhancing the security and intelligence and ensuring big data storage and efficient management of the IoV. Therefore, we expect that other innovative technologies such as Machine Learning, Big Data and 5G will facilitate its further development. To be more specific, the convergence of these technologies and BIoV is expected to result in generating innovative applications and services. Therefore, our purpose is to elaborate these technologies and propose research opportunities to accelerate their integration with BIoV

- Machine Learning with BIoV: Machine learning has been established as an efficient approach to support future BIoV. Machine learning, as a basis of artificial intelligence, has been used in numerous areas such as speech recognition, medical diagnosis, and computer vision. It has also revolutionized BIoV services as it enabled them to learn from training data and make data-driven conclusions, provide decision support, and predict the improvements of network performances. Thus, interdisciplinary research should focus on the integration of machine learning and BIoV, particularly regarding designing smart agents and learning-based analysis of the Blockchain-based IoV system. So-called smart agents have the capacity to manage the Blockchain and identify abnormal behaviors. The detection of abnormal behaviors is critical for the public chain, whereas the proper managing of the network is crucial for consortium chain and private chain because they need coordination among users. Furthermore, the learning-based analysis of the Blockchain-based system is not common. the traditional centralized systems do not offer enormous data that is available for evaluating the performance of the decentralized Blockchain structure. However, learning-based analysis can reveal important information about the mechanism design of the Blockchain structures as well as on-time forecasting models: First, Blockchain should support anonymous data sharing. Users are increasingly interested in privacy issues because of the growing number of IoT and wearable devices. In combination with data fusion, it is possible to develop multiple layer Blockchain structures that include sophisticated data authorization for different users. Second, The Blockchain mining activity represents an MDP process.

CONCLUSION

Several studies aimed at determining the optimal mining strategy via single-agent reinforcement learning. However, pool mining is still prevalent in comparison to individual mining. More precisely, pool mining is performed by miners who collaborate but at the same time also compete in mining blocks. A multi-agent reinforcement learning (MARL) involves a mixed setting of collaborative and competitive agents. Therefore, it can model the complex pool mining activity and allow the miners to determine optimal mining strategies. Cryptocurrency has a critical role in the public chain. Various chains use different cryptocurrencies. Cryptocurrencies and cryptocurrency portfolios have been established as investment options comparable to traditional financial products. Several studies have investigated cryptocurrency price prediction via supervised learning techniques. However, the potentials of Reinforcement learning (RL) or deep RL have not been fully researched. Importantly, RL and deep RL can have outstanding performance in regard to financial forecasts such as stock price prediction, considering that historical data do not reflect the current market conditions accurately. Consequently, this leads to poor prediction performance of changes in future price. Thus, it is recommended to adopt RL, deep RL, or inverse RL to investigate the investment return of cryptocurrencies.

- Big Data with BIoV : Owing to the significant progress of BIoV applications, big data has become a critical data analytic tool for using the potential of information located in enormous Blockchain IoT data. In future networks, it is assumed that BIoV will undergo exponential growth of data traffic in terms of the variety, velocity, and volume of Blockchain data. Big data can support many solutions to facilitate BIoV systems, including analytics, data cleaning, and storage [284]. In addition, big data facilitates cleaning services considered as a pre-processing phase that occurs prior to big data analytics. In this pre-processing phase, the integration and the improvement of the quality of big data are performed. More precisely, the cleaning service occurs in two primary phases. The first phase is data integration, which is also called data aggregation or data fusion. It is followed by data quality management whose purpose is to detect low-quality data, such as redundant data or corrupted data in BIoV data collection services (e.g., Blockchain-based sensor networks). Furthermore, the analytics service consists of the methods and models for data analysis and processing (e.g., MapReduce processing and data clustering algorithms) [285]. Data clustering is commonly employed to characterize the use and performance characteristics of big systems that are based on peer-to-peer consensus. More precisely, Blockchain datasets (e.g., Bitcoin data) are aggregated and subsequently analysed and visualized in order to identify unanticipated patterns in Blockchain networks. Furthermore, BIoV

CONCLUSION

supports big data regarding enhanced protection of privacy and data integrity to ensure secure storage of data analytics in big data. Under these conditions, Blockchain is an excellent solution for problems related to big data [286]. The decentralized management ensured reliability and authentication of Blockchain, which in turn, ensures the security of big data resources. Particularly, Blockchain ensures a transparent and trustworthy exchange of big data among users and service providers. By resolving security bottlenecks, BIoV can support universal data exchange on a large scale. Researchers have recently developed several big data models that use Blockchain, including data tracing solutions using Blockchain transactions [287] and data sharing with smart contracts [288]. According to preliminary results, big data can contribute to Blockchain-based IoV solutions in regard to performance and security.

- BIoV in 5G Networks and Beyond: The next generations of mobile networks (5G and beyond) have profoundly reshaped industries and entire societies. Its innovation lies in critical advantages such as enormous data connectivity, high system throughput, minimized operational costs, preservation of energy, low network latency, and high data rate. Furthermore, new technology architectures in 5G wireless networks such as cloud computing, device-to-device (D2D) communications, network slicing, network functions virtualization (NFV), and software-defined networking (SDN) have also brought new security issues [289]. To illustrate, security issues such as the lack of trust mechanisms between controllers and management applications, attacks on vulnerabilities in controllers, control plane communications, and switch, and faked and forged traffic flows are typical for SDN [290]. Furthermore, it is still challenging to ensure the integrity of platforms and service providers and prevent data leakage risks in resource sharing among NFV users and servers [291]. The Blockchain can provide feasible security solutions to such problems. To illustrate, Blockchain can be used to establish decentralized authentication mechanisms for SDN. Accordingly, the implementation of decentralized access authorization with smart contracts will be enabled [292]. Blockchain can also employ shared ledgers to establish trust among network entities, such as network users and SDN controllers, and to ensure secure data exchange and communication. Blockchain can also be used in NFV to allow for providing network functions and guaranteeing system integrity against data threats such as data attacks and malicious VM modifications [292]. Furthermore, 5G uses the concept of network slicing to support IoT applications. Network slicing allows several users to use the same physical hardware. Accordingly, the network slicing operation has a disadvantage of inter-slice security problems. To illustrate, if several

CONCLUSION

slices share the communication link, a malicious user on one slice can adversely impact other slices by compromising the data of the target slice or exploiting the resources [293]. In such a situation, Blockchain can be used to establish reliable end-to-end network slices and facilitate managing of resources by network slice providers [294]. Blockchain uses smart contracts to ensure authentication upon receiving a request by a slice provider to establish an end-to-end slice. Accordingly, the resource providers carry out resource trading on contracts with sub-slice components. In this process, Blockchain is used to immutably record and store the information of sub-slice deployment. Considering D2D communications in 5G networks, Blockchain establishes trust between D2D users and enables them to exchange data in a reliable and transparent manner [295]. In the Blockchain-based D2D scenario, edge servers and resourceful devices such as powerful smartphones and laptops take part in the Blockchain mining process. In contrast, lightweight D2D devices do not require Blockchain mining, but simply join the network for communication purposes [296]. Furthermore, Blockchain supports 5G services properly. To illustrate, Blockchain achieves trust management for 5G mobile vehicular communication owing to immutable and decentralized features [297]. The 5G-VANET scheme employs Blockchain to identify network attacks and deter data threats and do not allow them to enter vehicular ecosystems. Moreover, Blockchain is used to ensure flexible and secure key management in 5G IoT networks [298] by increasing communication security and minimizing computational complexity. Blockchain in combination with cloud computing has significant advantages that can be used in 5G network management. To illustrate, the Blockchain is used to establish reliable end-to-end network slices and facilitate the management of resources by network slice providers. The authors of [299] employed Blockchain to perform dynamic control of vehicle-to-vehicle and vehicle-to-everything communications in vehicular network slices. Furthermore, the cloud-native architecture is used to enhance 5G network slicing functions owing to its programmable networking. For example, the authors of [300] confirmed that owing to the cloud-native model, life-cycle slice management can generate, organize, and optimize network slice performances regarding data throughput, end-to-end delay, and resources. These results provide significant insights for the next generation of the BIoV-5G networks.

Extended Summary in French

1. Introduction

Eu égard à la croissance rapide du nombre des véhicules au cours des deux dernières décennies et en dépit des améliorations notables apportées à l'infrastructure, les solutions de transport mises en place n'arrivent plus à répondre à la croissance perpétuelle du trafic routier de nos jours. Le besoin d'intégrer des Systèmes de Transport Intelligents (STI) est désormais crucial. En effet, les STI visent à réduire les problèmes de congestion routière, améliorer l'efficacité du trafic routier et contribuer au développement des routes intelligentes. A cet effet, les utilisateurs reçoivent des informations pertinentes sur la disponibilité des places et sur les autres conditions du trafic, ce qui est susceptible d'optimiser la sécurité et de réduire le temps de déplacement. Grâce au développement rapide des technologies informatiques et de communication innovantes, le concept original des réseaux ad-hoc de véhicules (VANET) s'est transformé en un nouveau concept, à savoir l'Internet des Véhicules (IoV) [1–3]. L'IoV est un prérequis nécessaire pour les STI car il permet la connexion des véhicules intelligents à internet. L'IoV permet l'interconnexion des véhicules intelligents avec les infrastructures routières et les piétons pour répondre aux exigences fonctionnelles de plus en plus complexes des STI et favoriser le paradigme de véhicule-à-tout (V2X). Cependant, le nombre croissant des véhicules intelligents et des applications et des services véhiculaires connexes créera inéluctablement d'énormes quantités de données et de trafic réseau. De surcroît, la complexité des caractéristiques et du contexte de l'IoV ainsi que sa faible latence et sa forte mobilité comporteront des défis inhérents à la sécurité, la gestion et le stockage cloud. Récemment, la technologie de la Blockchain [13, 14] a manifesté un potentiel pour l'optimisation des Systèmes de Transport Intelligents tout en assurant leur sécurité, leur distribution et leur autonomie. Il convient ainsi d'opter pour une exploitation plus efficace de l'infrastructure et des ressources des STI telles que la technologie de la production participative (crowdsourcing). En effet, la Blockchain s'avère

cruciale pour relever les défis liés à la confidentialité et à la sécurité des réseaux IoV. Pour surmonter ces défis, cette thèse propose des solutions basées sur la Blockchain pour établir un paiement et une communication sécurisés en vue d'étudier l'utilisation de la Blockchain en tant que middleware entre les différents participants des STI. Le Framework proposé emploie l'Ethereum (ETH) pour développer une solution qui vise à faciliter les communications et les paiements Véhicule-à-tout (V2X).

2. Solutions proposées

Le présent chapitre présente de manière détaillée des solutions Blockchain développées dans le contexte d'IoV. Nous expliquerons l'architecture et les différents composants logiciels et matériels des deux solutions et nous présenterons des scénarios nominaux élaborés. De plus, nous introduirons un Framework basé sur un modèle pour valider l'approche suggérée. Ce Framework est principalement basé sur l'utilisation des formalismes des Arbres d'Attaque (AT) et des Automates Temporisés (TA) utilisés pour tester les aspects fonctionnels, de charge et de sécurité. La première section introduit la solution Internet des objets (IdO) pour la communication des Véhicules (DISV) visant à faciliter l'échange des données et la coopération entre les véhicules, l'infrastructure et les autres acteurs des STI. La DISV permet aux utilisateurs des réseaux internet des véhicules de recevoir des messages et de les diffuser sur la couche Blockchain de leur choix. Par ailleurs, le serveur confirme la réception du block en fonction de sa connaissance locale et il décide s'il doit être ajouté au contrat intelligent (Ethereum Smart Contract). La deuxième section introduit le Framework Blockchain « PSEV » qui permet la communication des véhicules et le paiement du parking. PSEV comprend les deux modules suivants : PSEV-Payment et PSEV-Communication. PSEV-Payment est une solution Blockchain utilisant Ethereum pour générer un système de paiement intelligent pour les parkings, tandis que la solution Blockchain de l'IoV pour la communication des véhicules (PSEV-Communication) fonctionne en temps réel pour assurer une communication sûre entre tous les participants des systèmes de transport.

2.1. Solution Blockchain de l'IoV pour la communication des véhicules (DISV)

Ce travail a pour objet de présenter une solution IoV avec une Application en Temps Réel (ATR). Cette solution fournit une communication sécurisée entre les véhicules et les autres acteurs dans les systèmes de transport. Elle tente de surmonter les lacunes telles que le délai d'exécution et d'améliorer la performance. Un prototype DISV a été développé et il a testé la solution selon les scénarios

suivants : si le conducteur est somnolent, les voitures à proximité doivent être alertées en envoyant un message via la Blockchain. Etant basée sur une architecture IdO, la solution proposée devrait contenir principalement trois couches, à savoir la perception, le réseau et l'application qui sont décrites ci-dessous :

- La couche de perception : Il s'agit de la couche physique qui consiste en plusieurs appareils IdO équipés de capteurs conçus pour identifier et recueillir les informations relatives à l'environnement (c.à.d. les paramètres physiques) et pour détecter les objets intelligents à proximité. L'application Android pour les véhicules (AV) embarquée dans la couche de perception collecte et analyse les données sur le trajet, le véhicule et le comportement du conducteur. L'application Android pour les infrastructures (AP) stimule le rôle des appareils IdO intégrés dans les routes tels que les radars, les feux de signalisation, les signes électroniques routiers, etc.
- La couche du réseau : Elle connecte les capteurs avec les autres serveurs, les appareils réseaux et les objets intelligents. De plus, elle transmet et traite les données relatives aux capteurs.
- La couche d'application : Elle comprend une sous-couche Blockchain et un serveur central cloud. Elle délivre des services spécifiques à l'application pour les services IdO. Plus précisément, l'application Blockchain gère la communication entre les véhicules et les autres acteurs dans les systèmes de transport. Le Serveur Central Cloud est chargé de traiter et d'analyser les données obtenues et de gérer les invitations des autres acteurs.

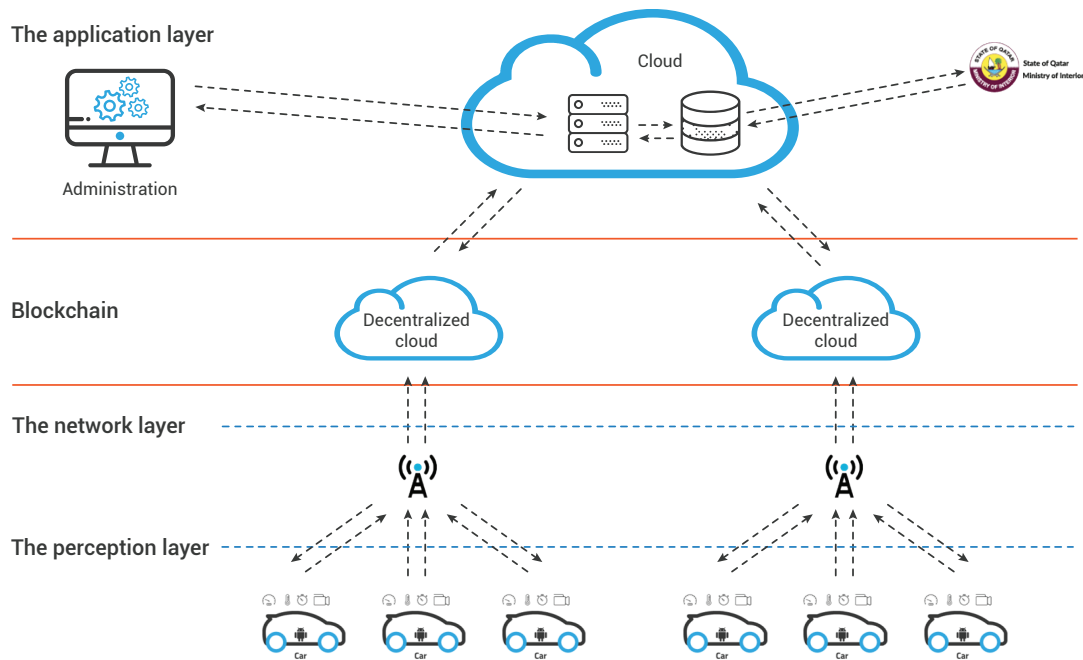


Figure S1: L'architecture de la solution IdO proposée.

La figure S1 illustre l'architecture de la solution proposée et montre le flux de travail principal qui comprend trois étapes principales. Tout d'abord, les voitures envoient des données au serveur central (1). Ensuite, en fonction des données reçues, le serveur central envoie une invitation pour se connecter à la couche de la Blockchain (2). Puis, les voitures peuvent partager les données avec les autres participants de l'IoV de la même zone en toute sécurité (3).

2.1.1. La couche de perception

Pour tester les scénarios possibles en impliquant les différents composants, une application Android a été développée dans l'application Android pour les Véhicules (AV) et pour l'Infrastructure (AP) comme détaillé dans les sections qui suivent.

Application Android pour les Véhicules (AV)

AV est une application Android qui comprend deux sous-systèmes. Le premier sous-système est le Système de Collecte des Données du Véhicule (VDCS) qui recueille les données relatives au trajet et à la voiture. Le deuxième sous-système est le système de détection de la somnolence du conducteur

qui recueille les données relatives au comportement du conducteur pour identifier s'il est somnolent ou pas :

1. Le VDCS est conçu pour recueillir des informations sur la voiture telles que le modèle de la voiture et les caractéristiques du moteur y compris sa puissance, sa vitesse et sa taille. Finalement, le système recueille les informations relatives au trajet telles que le temps de début et de fin, la distance et la vitesse minimale, moyenne et maximale tel qu'illustré dans la figure S2. Il est configuré pour détecter certaines mesures comme la vitesse de rotation des axes lacet, tangage et roulis ; l'accélération ; la distance et la localisation GPS toutes les 15 secondes.

2. La détection de la somnolence du conducteur a pour objet de détecter la somnolence du conducteur et de prévenir les accidents potentiels susceptibles de survenir. Ce système est un élément du Système Avancé d'Assistance au conducteur (ADAS) qui est partie intégrante de la technologie automobile contemporaine. L'ADAS a pour rôle l'amélioration de la sécurité et de l'expérience de conduite. Ce système a été développé avec la détection en temps réel de la somnolence du conducteur à travers les techniques de réseaux de neurones profonds. Vous pouvez trouver plus de détails sur ce système dans les pages [45–47] L'application Android comporte pratiquement 4 pages. La première page est la page d'authentification où il faut utiliser un nom d'utilisateur et un mot de passe. Une fois authentifié, l'utilisateur peut démarrer un nouveau trajet ou accéder aux informations liées aux cinq derniers trajets dans la deuxième page. Si l'utilisateur choisit un nouveau trajet, l'application enregistrera et affichera toutes les informations comme détaillé dans la section précédente. Ensuite, elle enverra les données recueillies via le service web au serveur cloud. Dans la quatrième page, la caméra frontale capturera et affichera le visage du conducteur.

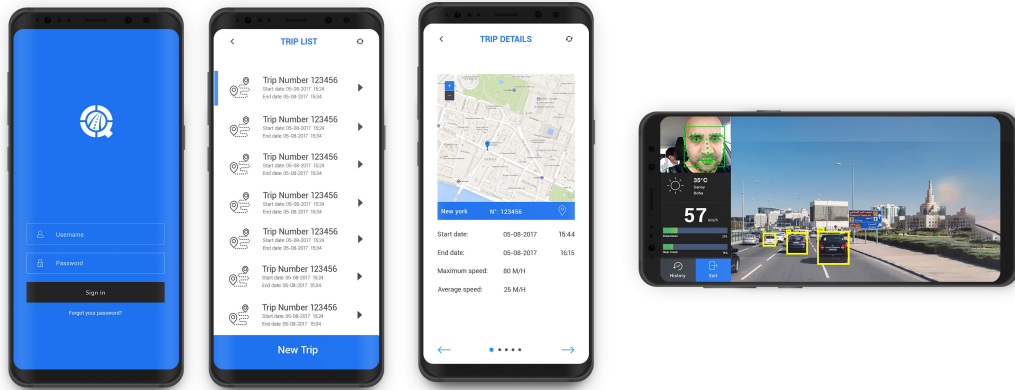


Figure S2: Capture d'écran des 4 pages de l'application Android pour les Véhicules (AV).

Application Android pour l'Infrastructure (AP)

L'objectif de cette application est de stimuler le rôle des appareils IdO embarqués dans les routes tels que les radars, les feux de signalisation, les signes électroniques routiers, etc. Il est possible d'ajouter d'autres options à la couche de perception de l'application Android telles que les circulations, les vitesses des voitures et les conditions météorologiques.

2.1.2. La couche du Réseau

La couche du réseau établit une connexion entre les serveurs, elle transmet et elle traite les données des capteurs. L'application peut utiliser le réseau Wifi ou l'internet mobile (3G/3G+/4G/5G) pour envoyer les données au serveur. Ce processus de collecte utilise un système hybride pour rassembler et stocker les données avant de les transmettre au serveur. Cette technique a fait preuve d'efficacité pour la collecte des données lorsque la connexion internet est faible ou instable.

2.1.3. La couche d'application

La couche d'application contient deux principaux composants : le serveur central cloud et le système de communication utilisant un réseau Blockchain.

Le serveur central cloud

Le serveur central cloud délivre des services spécifiques à l'application à l'utilisateur final. Il envoie les données collectées aux services web les traiter et les analyser avant de les montrer à l'utilisateur final. Le service web est un composant de la couche de l'application responsable de l'interaction entre les différents composants de la solution proposée, tels que les sites web, les serveurs des bases de données et les systèmes embarqués. La Windows Communication Foundation (WCF) établie par Microsoft est utilisée pour mettre en place le service web en se basant sur l'architecture REST et le format de message JSON. En plus de la collecte des données des appareils, elle utilise les informations relatives aux accidents de la Direction Générale du Trafic relevant du Ministère de l'Intérieur ainsi que les conditions routières et toutes les données pertinentes provenant des autres autorités. Les données sont disponibles à l'utilisateur final à travers le site web avec un accès direct aux services web. Le site web est l'interface utilisée par les chercheurs pour interagir et demander les données enregistrées. Le site web affiche des informations démographiques sur la nationalité du conducteur, son genre et son âge. Il comprend également des informations sur le véhicule, le modèle et la date à partir de laquelle il a été mis en service.

La sous-couche de la Blockchain

La sous-couche de la Blockchain gère la communication entre les voitures. Dans chaque créneau horaire distinct, la voiture envoie les données collectées au serveur central via le service web. Les données comprennent la localisation actuelle et l'état de connexion de l'une des couches existantes de la Blockchain. Ensuite, le serveur central invite les appareils IdO proches pour établir la communication à travers une Blockchain cloud disponible. Ainsi, la communication est établie après avoir accepté l'invitation. Chaque section routière contient une couche Blockchain qui envoie les messages aux appareils IdO connectés. La sous-couche de la Blockchain et l'application Android créent conjointement des applications décentralisées, telles que Dapp, dApp ou DApp. Plus précisément, les applications décentralisées représentent les applications internet distribuées exécutées sur un réseau décentralisé pair-à-pair (Blockchain). Les applications Dapp ne dépendent pas d'un serveur central comme est le cas pour les applications standard. La sous-couche de la Blockchain constitue l'extrémité-arrière (Back-end) des applications décentralisées, tandis que l'application Android constitue l'extrémité-avant (Front-end). L'application mobile appelle les fonctions du contrat intelligent

déployé dans chaque nœud Ethereum pour envoyer un message à travers le réseau de la Blockchain. La communication passe comme un passerelle (gateway) entre le téléphone mobile et l'extrémité du nœud. Cette étude utilise l'un des Frameworks les plus fiables, à savoir le Framework Android Web3.Js. Son contrat intelligent comprend deux fonctions principales. La première fonction "SetMessage" se charge de la publication d'un nouveau message sur le réseau de la Blockchain en fonction de la quantité d'ETH que l'expéditeur désire payer par unité de gaz pour extraire le message. La deuxième fonction "GetMessage" permet à l'appareil de se connecter au réseau de la Blockchain pour lire les données existantes.

2.2. Solution Blockchain de l'IoV pour le paiement (PSEV)

Cette solution proposée assure une communication sécurisée entre les véhicules et les autres acteurs des systèmes de transport et elle permet l'accomplissement des paiements. Elle tente de surmonter les lacunes telles que les longues durées d'exécution ce qui est susceptible d'optimiser la performance. Etant basée sur l'architecture IdO, la solution proposée se compose de trois couches : la perception, le réseau et l'application, tel que décrit ci-dessous :

- La couche de perception – Cette couche se compose de trois solutions développées sous forme d'applications Android, à savoir Application Android Auto pour les Véhicules, Application Android pour les Locataires des Places de Parking et Application Android pour le Système de parking IdO. A travers ces trois applications, l'utilisateur peut établir une communication, gérer sa place de parking et effectuer des paiements.
- La couche du réseau – Cette couche utilise les réseaux sans fil pour connecter la couche de l'application aux autres appareils, tels que les capteurs du parking et des véhicules.
- La couche de l'application – Cette couche héberge le serveur et les applications responsables du système de communication et de paiement et elle répond aux besoins particuliers du système de gestion des appareils intégrés dans la solution Blockchain.

La figure S3 visualise ladite architecture et l'explique en plus de détails dans le schéma ci-dessous.

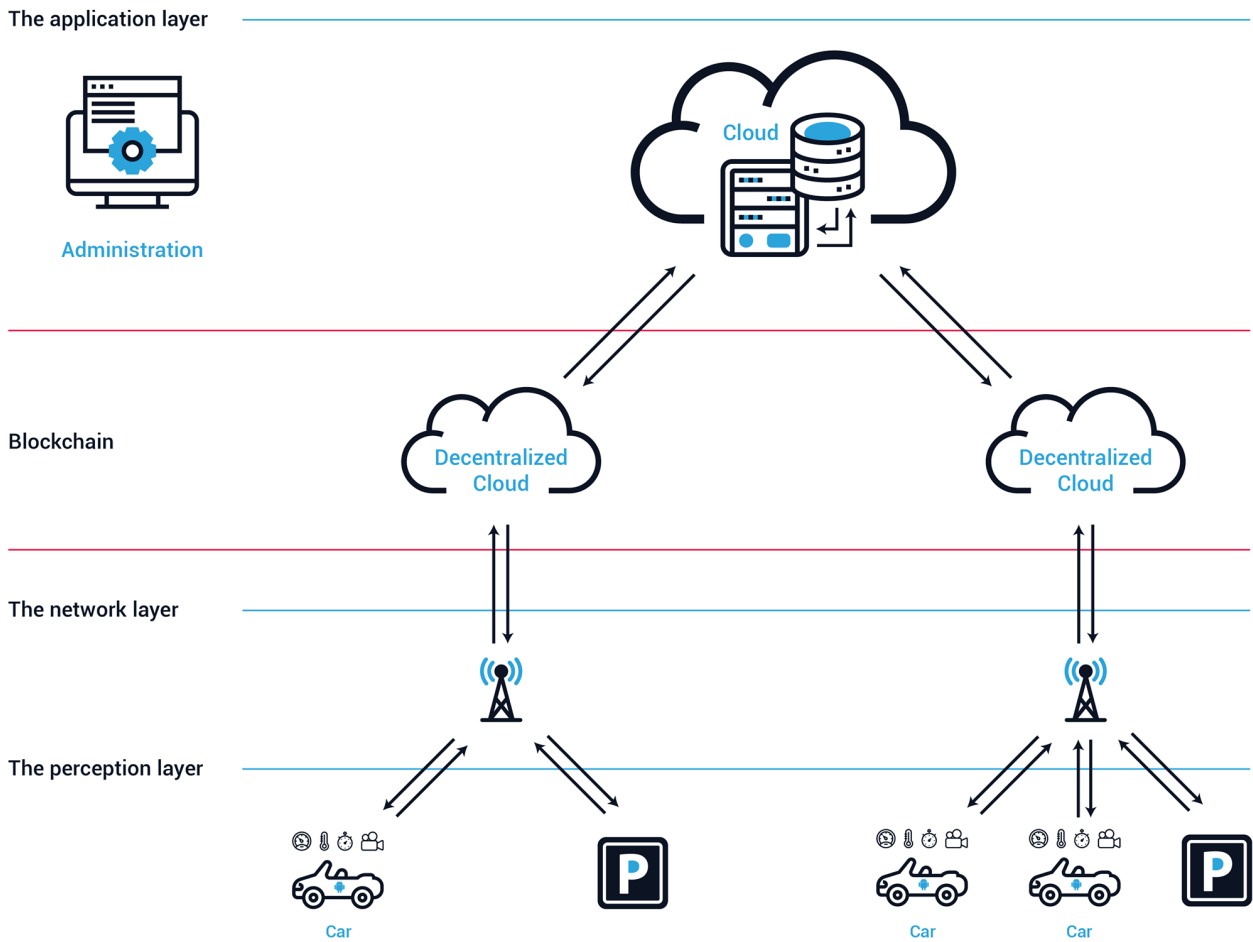


Figure S3: L'architecture Internet des véhicules développée PSEV.

2.2.1. La couche de perception

Pour tester les scénarios possibles y compris les différents composants, une application Android a été développée dans l'application Android pour les Véhicules (AV) et pour l'infrastructure (AP) comme détaillé dans les sections suivantes.

Application Auto Android pour les Véhicules

Il s'agit de l'interface de la voiture employée par l'utilisateur pour envoyer et recevoir des messages, chercher, naviguer et payer pour le parking. De plus, cette application détecte la proximité de la voiture par rapport à la station de paiement du parking pour initier la transaction Blockchain. Le contenu de l'application Android est illustré dans la figure S4 et la figure S5.



Figure S4: Capture d'écran de l'interface de l'application Android à l'intérieur du véhicule.

EXTENDED SUMMARY IN FRENCH

Comme montré dans l'image précédente, l'interface constitue la page d'authentification qui s'affiche lorsque l'application est démarrée. Ensuite, l'écran affiche la carte avec les parkings disponibles. Lorsque l'utilisateur sélectionne le parking, l'application fournit les informations de navigation. Une fois que l'utilisateur arrive au parking, l'application affiche la liste des places de parking disponibles à l'endroit sélectionné et leurs plages horaires. Lorsque l'utilisateur sélectionne la plage horaire qui lui convient, l'application lance le chronomètre et affiche les coûts. Une fois que l'utilisateur désire quitter la place de parking, il arrête le chronomètre en cliquant sur "STOP". En ce moment, l'application affiche les détails de la transaction, tout en sachant que l'utilisateur peut accéder à l'historique à tout moment pour vérifier tous les détails des transactions précédentes.

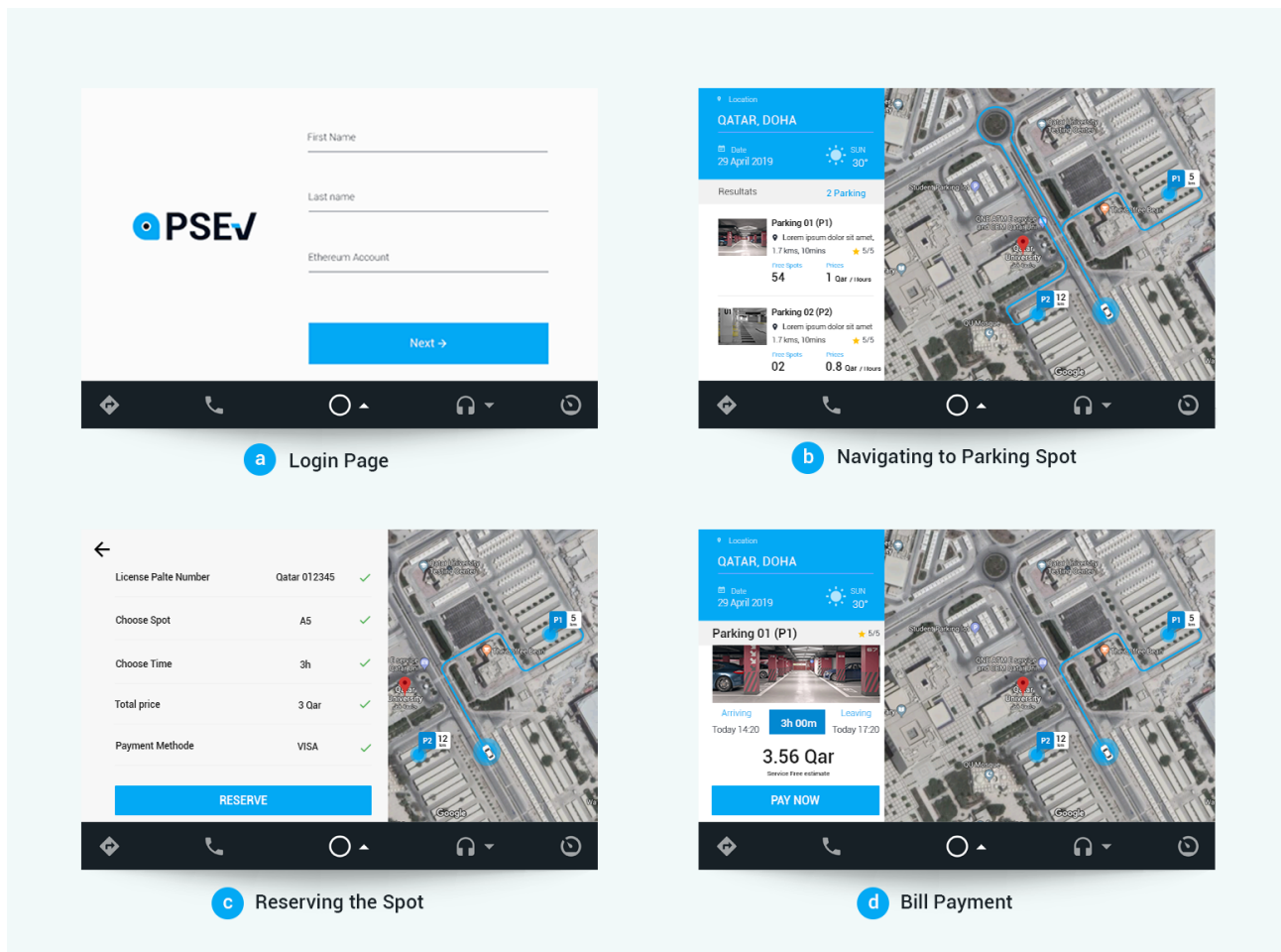


Figure S5: Capture d'écran des principales pages de navigation de l'application Android.

Application Android pour les Locataires des Places de Parking

Cette application Android permet au propriétaire de l'immeuble/parking de gérer et de définir les plages horaires dans lesquelles leurs places de stationnement sont disponibles à la location au public, lorsqu'ils ne sont pas utilisés. Ainsi, les propriétaires du parking peuvent générer des revenus des places de parking non-utilisées. Cette application contient principalement 3 pages. La première constitue la page d'authentification. La deuxième représente la page de configuration de la place de parking qui permet à l'utilisateur de définir les plages horaires auxquelles sa place de parking est disponible pour la location. Les propriétaires peuvent accéder aux listes des transactions précédentes le cas échéant, et voir le montant total perçu de la place de parking. La troisième page est dédiée à la gestion du compte et aux tâches administratives.

Application Android pour le Système de parking IdO

Il s'agit d'une application embarquée dans la machine de paiement du parking IdO qui détecte les voitures proches dans la Blockchain et qui donne l'accès aux places de parking aux voitures qui les avaient sollicitées. Le portail de paiement ne sera pas limité à l'authentification et à la gestion des plages horaires, mais il gère aussi la transaction entre la voiture et le fournisseur de la place de parking. Cette application peut jouer le rôle de la machine d'accès au parking qui garantit l'accès physique à la place de parking. Toutes les applications ci-dessus sont des acteurs principaux dans le système de la Blockchain et toutes les transactions et les messages envoyés sont sécurisés en vertu des contrats de la Blockchain.

2.2.2. La couche du réseau

La couche du réseau vise à établir une connexion entre les portails de paiement du parking localisés dans la couche de perception (Application Android Auto pour les Véhicules, Application Android pour les Locataires des Places de Parking et Application Android pour le système de parking IdO), les voitures intelligentes et le serveur central cloud dans la sous-couche de la Blockchain. Cette couche transmet et traite les données IdO en toute sécurité via le réseau Wifi ou l'internet mobile (3G/3G+/4G/5G).

2.2.3. La couche d'application

Le serveur central cloud et le système de communication et payments sont les principaux composants de cette couche. Ils fonctionnent à travers le réseau de la Blockchain.

Le serveur central cloud

L'application de gestion administrative est hébergée dans le serveur central cloud. Le cloud surveille les systèmes et le fonctionnement de l'infrastructure à travers le rôle d'administrateur. Ainsi, il dispense des services administratifs et d'hébergement pour tous les portails de paiement et les voitures qui participent dans le système. L'utilisateur emploie l'Application Android Auto pour les Véhicule pour payer le ticket. Ensuite, le serveur central invite les appareils IdO de la même zone à communiquer entre eux à travers l'une des instances disponibles du cloud de la Blockchain. Plus précisément, les appareils IdO reçoivent une invitation pour communiquer via la sous-couche de la Blockchain. Il existe une application web dans le cloud central qui sert d'outil dans la couche de l'application employée pour établir une interaction avec plusieurs éléments de la solution IdO, y compris le site web administratif, le serveur de la base de données des appareils qui interagissent avec la Blockchain et les systèmes embarqués pour les portails de paiement. Le service cloud de Windows Azure et les appels de REST API ont été utilisés pour héberger ce service. Plusieurs outils scientifiques pouvant analyser le flux du trafic dans les différentes places de parking ont également été créés dans cette application web afin d'étudier et de comprendre les exigences relatives aux demandes de stationnement dans les différentes localisations de la ville. Il s'agit ainsi d'une plateforme administrative centrale à utiliser par l'administrateur pour dépanner le système le cas échéant et assurer un fonctionnement régulier.

La sous-couche de la Blockchain

La Blockchain facilite la communication entre les différents éléments et entre les utilisateurs des routes des villes intelligentes et les paiements entre les voitures et les fournisseurs de parking. Plus précisément, à chaque fois qu'un utilisateur envisage effectuer un paiement pour le parking, cette transaction est diffusée sur la sous-couche de la Blockchain, puis validée. Les transactions accomplies font désormais partie du block si bien qu'ils deviennent valides et immutables. La sous-couche de la Blockchain et l'application Android sont fusionnées pour générer des applications décentralisées (Dapp, dApp ou DApp). Les applications décentralisées sont des applications internet distribuées opérant sur

un réseau décentralisé pair-à-pair tel que la Blockchain. L'application Android constitue l'extrémité frontale (Front-end) tandis que la sous-couche de la Blockchain constitue l'extrémité arrière (Back-end) des applications décentralisées. Les dispositions du contrat intelligent sont déployées dans tous les nœuds de l'Ethereum et l'application mobile utilise la Blockchain pour envoyer des messages. Le téléphone mobile et les extrémités des nœuds établissent une communication. Cette solution emploie exceptionnellement le Web3.Js qui est fiable pour le framework d'Android.

La sous-couche de la Blockchain est équipée du système suivant :

- **Système de gestion d'accès :** Le système de gestion d'accès permet l'identification et l'authentification des différents participants des systèmes de transport intelligents pour échanger les données. Les modules suivants sont localisés dans la sous-couche de la Blockchain dans le système de gestion d'accès :
 - **Module de gestion du véhicule :** Le contrat de gestion du véhicule est un élément crucial pour ce module. Il permet l'ajout, la modification et la suppression du véhicule.
 - **Module de gestion du parking :** Le contrat de gestion du parking est également un élément important pour ce module. Il permet l'ajout, la modification et la suppression du parking dans le système.
 - **Module de gestion du participant :** Le contrat de gestion du participant permet l'ajout, la modification et la suppression d'un participant du système de transport intelligent. Ce dernier peut envoyer des messages au système par rapport aux routes, aux signes électroniques, aux feux de signalisation et aux radars.
- **Système de gestion du paiement du parking:** Ce système permet la gestion du parking à travers deux types de contrats : un contrat intelligent qui permet aux fournisseurs de vendre des tickets et d'avoir les numéros des tickets disponibles et un contrat intelligent qui permet aux clients ou aux véhicules d'acheter les tickets disponibles.
- **Système de gestion de la communication:** Le contrat intelligent inhérent à la gestion de la communication a deux fonctions principales. La fonction "Envoyer un message" qui transmet un nouveau message au réseau de la Blockchain. Cette action est basée la quantité d'ETH que l'expéditeur désire donner par unité de gaz pour extraire le message. La fonction "Lire le

message” permet au participant d’accéder aux données à travers les appareils qui font partie du réseau de la Blockchain.

3. Evaluation de la performance et discussion

3.1. Solution Blockchain de l’IoV pour la communication des véhicules (DISV)

La performance de la solution logicielle peut être évaluée à travers différentes méthodologies [49, 5]. Il est nécessaire d’évaluer les propriétés spécifiques requises pour le fonctionnement régulier de la solution. Les principales propriétés de la solution proposée sont : le temps d’exécution, les coûts, la disponibilité, l’intégrité, l’immuabilité et la sécurité. Dans ce résumé, on se focalisera sur le coût, le temps d’exécution et la mémoire des propriétés de consommation de l’énergie.

Coûts

Le Testnet du réseau de l’Ethereum a été utilisé pour déployer le prototype du contrat intelligent. Dans cette section, les coûts de création et d’exécution du contrat intelligent sont analysés. Les valeurs suivantes qui ont été validées en janvier 2020 ont été utilisées : 1 gas_1 wei (0.000000001 ETH) et 1 ETH 161,92 USD. La valeur minimale du gaz à utiliser dans la transaction a été définie à 1 wei. La valeur moyenne approximative de gaz est de 0,006845 Ethereum en ce moment d’analyse. 1 Gaz = 0,006845 Ethereum (ETH) Prix du gaz = 6,138,887 Gwei Le tableau S1 présente les coûts d’exécution des différentes fonctions de l’application. En effet, la création et l’adoption du prototype sur la Blockchain coûte cher, soit 0,07572 USD. Cependant, il convient de mentionner qu’il s’agit d’un coût unique pour configurer et lancer le système. De plus, ce coût peut être minimisé en supprimant le Framework Truffle qui est utilisé pour l’établissement, l’adoption et la gestion des contrats intelligents. Il comprend également des fonctionnalités telles que le contrat de “Migrations” pour gérer le cycle de déploiement.

Table S1: Le taux du coûts des différentes fonctions du contrat intelligent basé sur 1 ETH = 161,92 USD et 1 gas = 0,000000001 ETH .

Fonction	Gaz utilisé	Prix
Deploy Contract	389,473	0.07572 (one time / Truffle)
SendMessage	140,345–257,488	0.02486–0.0456
Get Message	0	0

En revanche, la fonction “SendMessage” n’est pas chère, elle coûte 0.03523 USD en moyenne. Néanmoins, il existe des variations significatives sur les coûts des fonctions “SendMessage” à cause de la variabilité des tailles des messages d’insertion. Toutefois, le prix de l’octet est défini à 136 gaz (\$0,00003 USD). Par contre, la fonction “GetMessage” ne requiert aucun coût supplémentaire car l’extraction n’est pas nécessaire lors de la réception des messages des blocks d’autant plus que les mises à jour ne sont pas requises pour le contrat intelligent.

Temps d’exécution

Le temps d’exécution fait partie des critères d’évaluation des systèmes de gestion de transport comme DISV. En effet, même un petit retard dans l’envoi ou la réception des messages peut entraîner de graves perturbations sur le système. Pour garantir le bon fonctionnement du Framework afin d’assurer la communication entre les véhicules et les autres acteurs du système de transport à travers la technologie de la Blockchain, il est essentiel d’assurer l’ajout ponctuel de chaque message au contrat intelligent, étant donné que le processus d’extraction repose sur la résolution des problèmes complexes. Vu que le prototype proposé est une application en temps réel, le temps d’exécution s’avère très important. Dans les tests informatiques, les temps d’appel de chaque fonction de l’application Android sont mesurés. Pour évaluer la performance de la solution proposée par la Blockchain privée de l’Ethereum, un serveur avec une configuration de 64 GB de rame et un Core i7-000 ont été utilisés. La figure S6 montre que le temps de réponse du serveur de la fonction “GetMessage” est beaucoup plus court que le temps de réponse de “SendMessage”. En effet, l’appel à la fonction “GetMessage” prend entre 1 milliseconde et 10 millisecondes lorsque le serveur reçoit 1000 requêtes. Par conséquent, il n’y a de pas différence entre l’appel à la fonction “GetMessage” une fois ou cent fois en termes de temps d’exécution. Cependant, l’appel à la fonction “SendMessage” demande beaucoup de temps parce que le message doit être miné pour l’ajouter au contrat intelligent. Lorsque le serveur reçoit 10 requêtes

d'appel à la fonction "SendMessage", il prend 1,46 s et plus de 90 s en cas de 500 requêtes. Compte tenu des différents facteurs, l'étude informatique a montré qu'il n'est pas recommandé d'avoir plus de 25 messages dans la liste de "SendMessage". L'exécution en temps réel est une propriété principale

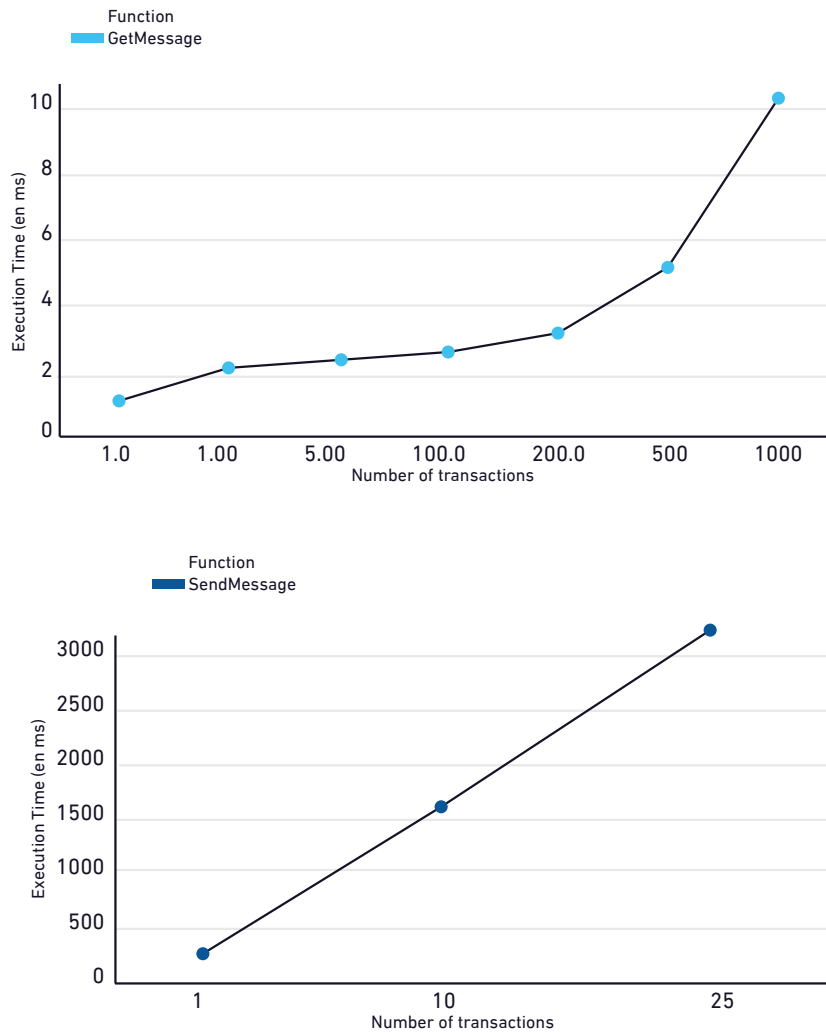


Figure S6: Temps d'exécution des différentes fonctions de la solution IdO décentralisée pour la communication des véhicules (DISV) en millisecondes.

de DISV étant donné que chaque transaction prend un temps d'exécution très court. Donc, il a été recommandé dans l'architecture suggérée d'utiliser la couche de la Blockchain dans chaque zone de sorte que le serveur reçoit et envoie un nombre réduit de messages. En plus d'utiliser plusieurs couches de la Blockchain, l'application Android supprime les messages anciens et dupliqués de manière à ce que le contrat contienne seulement les messages nécessaires. En utilisant l'architecture recommandée

en DISV, le temps de réponse du serveur de messagerie est généralement entre 0 et 3 s, ce qui fait de DISV une application en temps réel (RTA).

Consommation de la mémoire et de l'énergie

Etant donné que DISV utilise la Blockchain pour l'IdO, il est essentiel d'évaluer la consommation de la mémoire et de l'énergie, vu que les appareils IdO ont généralement des capacités énergétiques et informatiques faibles. Un modèle Huawei P8 Lite avec une configuration de 2 GB Rame, une batterie Li-Po 2500 mAh et un processeur Hisilicon Kirin 620 a été utilisé dans les tests informatiques de la démonstration. La figure S7 démontre que la consommation de la mémoire de la solution développée sous Android est beaucoup plus basse que celle des autres applications commerciales telles que Facebook (134 MB), WhatsApp (106) et Skype (233 MB). S'agissant de la consommation énergétique,

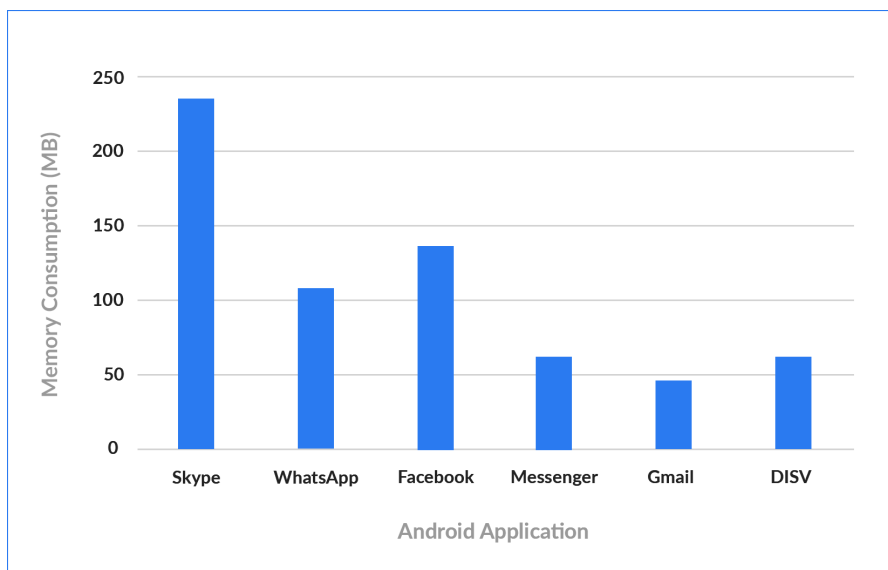


Figure S7: La comparaison de la consommation de mémoire du DISV avec des applications mobiles commerciales.

la solution proposée consomme une moyenne d'électricité de 23.43 mAh qui est similaire à celle de Skype et de Facebook qui consomment respectivement 21,66 mAh et 18,56 mAh, comme illustré dans la figure S8.

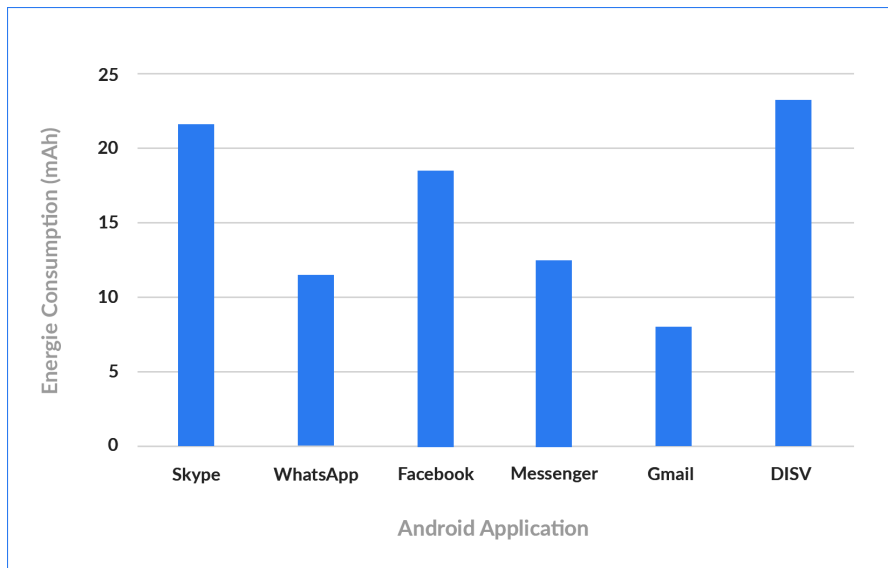


Figure S8: La comparaison de la consommation d'énergie du DISV avec des applications mobiles commerciales.

3.2. Solution Blockchain IoV pour le paiement (PSEV)

Il existe plusieurs méthodes pour évaluer la performance de la solution logicielle [273, 276].. En effet, l'évaluation des fonctionnalités particulières requises pour permettre le bon fonctionnement de la solution est encore plus important. Les caractéristiques principales de la solution proposée sont: l'intégrité, le coût, la cohérence, la confidentialité, le temps d'exécution, l'immutabilité et la consommation de la mémoire et de l'énergie. Le bon fonctionnement de l'opération exige que toutes ces caractéristiques atteignent des performances élevées. Dès lors, ce travail évalue les caractéristiques spécifiques pour évaluer la performance générale de la solution proposée. Dans ce résumé, nous nous focaliserons sur les propriétés inhérentes au coût, au temps d'exécution et à la consommation de la mémoire et de l'énergie.

Coût

Dans le déploiement standard de la Blockchain, les cryptomonnaies sont des incitations visant à récompenser les nœuds/ opérateurs du réseau, ce qui assure le consensus et l'intégrité des données ainsi que la décentralisation de l'écosystème. Néanmoins, compte tenu des coûts relatifs au stockage des informations et aux tâches informatiques, les utilisateurs de la Blockchain doivent payer pour

les incitations. Ainsi, ce travail examine les coûts liés aux services exécutés par DApp avec ces emplois. De plus, l'étude compare ces dépenses avec les coûts encourus dans les systèmes centralisés et propriétaires largement utilisés. Il est alors nécessaire d'estimer le coût relatif à la DApp lorsqu'elle fournit des services à plusieurs fournisseurs et clients. De surcroît, l'évaluation de la rentabilité est importante lorsque la Blockchain est utilisée pour substituer les éléments conventionnels des systèmes de parking en vue d'optimiser l'interopérabilité. Le Testnet du réseau de l'Ethereum a été utilisé pour le déploiement du prototype du contrat intelligent et pour évaluer la solution de la rentabilité. Cette partie tend à évaluer les dépenses requises pour créer et exécuter le contrat intelligent. Les taux suivants qui ont été validés en mai 2020 ont été utilisés : 1 gaz 1 wei (0.000000001 ETH) et 1 ETH 209 USD. La valeur minimale de gaz qui peut être utilisée dans une transaction est 1 wei. La valeur moyenne de gaz a été de l'ordre de 0.006252 Ethereum au moment de l'évaluation. 1 Gaz = 0,006252 Ethereum (ETH) Prix du gaz = 4,652,309.7157 Gwei Le tableau S2 donne un aperçu sur les coûts d'exécution des fonctions les plus fréquemment appelées. Comme expliqué, le coût maximal de 30,7 USD génère et déploie le prototype sur la Blockchain. Cependant, il convient de souligner qu'un seul paiement est requis pour établir et déployer le système. Par ailleurs, si Truffle est supprimée, ce coût s'élève. Truffle désigne le Framework utilisé pour établir, déployer et gérer les contrats intelligents. Il contient un contrat de Migrations pour gérer le cycle de déploiement. Notre analyse a révélé que "Add Vehicles", "Send Message", "Buy Ticket" et "Sell Ticket" sont fréquemment dénommés : fonctions. La fonction "Buy Ticket" coûte moyennement 0.028 USD. De plus, l'extraction n'est pas requise pour obtenir des messages des blocks lors de l'appel aux fonctions "Get Vehicles By Address", "Get All Vehicles", "Get Parking By Address", "Read Message" et "Get Tickets For Sale". En conséquence, les mises à jour ne sont pas nécessaires pour le contrat intelligent et ces fonctions n'engendrent pas des frais supplémentaires.

Temps d'exécution

Les tests sur la performance du système ont été menés en PSEV sur un serveur avec la configuration suivante : 8 GB Rame et Core i7-000. Le tableau S3 démontre que le temps de réponse du serveur des fonctions "GetTicketsForSale", "ReadMessage", "GetParkingByAddress", "GetAllVehicles" et "GetVehiclesByAddress" est beaucoup plus court que le temps de réponse des autres fonctions vu que l'extraction n'est pas requise pour interagir avec le contrat intelligent. Toutefois, l'appel à une fonction

Table S2: Les coûts des fonction de contrat intelligent.

Systeme	Fonction	Gas utilisé	Prix (US ~Dollars)
	Deploy System	0,14437926 ETH	\$30,70
La gestion des accès	Add Vehicles	132797	\$0.08446
	Get Vehicles By Address	0	0
	Get All Vehicles	0	0
	Add Parking	150346	\$0.09561
	Get Parking By Address	0	0
	Get All Parking	0	0
	La gestion de la communication	Send Message	89280
Read Message		0	0
La gestion des paiements de parking	Sell Ticket	166854	\$0.10613
	Get Number Of Tickets	0	0
	Get Tickets For Sale	0	0
	Buy Ticket	43078	\$0.2739

quelconque prend entre 2 et 157 millisecondes. Ainsi, l'évolutivité du PSEV avec une large population d'utilisateurs a été prouvée. Le temps d'exécution est un critère fondamental pour l'évaluation des systèmes de communications telles que PSEV-Communication. Plus précisément, les petits retards à l'envoi ou à la réception des messages peuvent entraîner des perturbations graves sur le système. Par conséquent, l'ajout instantané de tous les messages au contrat intelligent est nécessaire pour assurer un fonctionnement adéquat du Framework pour permettre la communication entre tous les participants du système de transport. Il convient de souligner que le processus d'extraction repose sur la résolution des problèmes complexes. La figure S9 démontre que l'appel à la fonction prend entre 2 millisecondes et 157 millisecondes s'il y a 1000 requêtes par serveur pour appeler la fonction ce qui ne crée pas une grande différence quant au temps d'exécution si la fonction a été appelée une fois ou même 100 fois. Par ailleurs, l'appel à la fonction nécessite plus de temps pour extraire le message et l'ajouter au contrat intelligent. 19 millisecondes sont requises lorsque 10 requêtes d'appel à la fonction "Send Message" sont reçues par le serveur et 2,748 secondes sont requises en cas de 1000 requêtes en PSEV, étant donné que le temps de réponse du serveur de la messagerie prend entre 0 et 3 secondes. Par conséquent, PSEV a été prouvé être une application en temps réel. Par ailleurs, les tests de performance indiquent que, par rapport à d'autres solutions similaires [277], l'application proposée est plus rapide et plus évolutive. Plus précisément, l'application proposée envoie jusqu'à 1145 messages en 3 secondes alors que les solutions similaires peuvent envoyer 25 à tous les participants dans les systèmes de transport

Table S3: Temps d'exécution de la plupart des fonctions demandées du contrat intelligent en millisecondes.

Systeme	Fonction	Temps (ms)
La gestion des accès	Add Vehicles	28
	Get Vehicles By Address	7
	Get All Vehicles	11
	Add Parking	32
	Get Parking By Address	9
	Get All Parking	15
La gestion de la communication	Send Message	4
	Read Message	2
La gestion des paiements de parking	Sell Ticket	43
	Get Number Of Tickets	18
	Get Tickets For Sale	12
	Buy Ticket	57

intelligents.

Consommation de la mémoire et de l'énergie

Vu que la PSEV utilise la Blockchain pour l'IdO, l'évaluation de la consommation de la mémoire et de l'énergie devient indispensable compte tenu des capacités informatiques et énergétiques faibles des appareils de l'IdO. Dans la démonstration, un modèle Huawei P8 Lite (Rame : 2 GB Rame, processeur : Hisilicon Kirin 620, batterie : Li-Po 2500 mAh) a été utilisé pour l'évaluation de performance. Les résultats présentés dans la figure S10 indiquent que la solution Android proposée a une consommation mémoire plus basse que les différentes applications commerciales comme Skype (233 MB), Whats-App (106 MB) et Facebook (134 MB). S'agissant de la consommation énergétique, la solution développée consomme en moyenne 25.43 mAh, ce qui est comparable à l'application Facebook et à l'application Skype, en utilisant respectivement 18,56 mAh et 21,66 mAh, comme le montre la figure S11.

3.3. Discussion

Cette section présente une comparaison entre la solution développée PSEV et DISV. Dans cette étude, nous avons comparé les solutions en termes de coûts, de temps d'exécution et de consommation de mémoire et d'énergie.

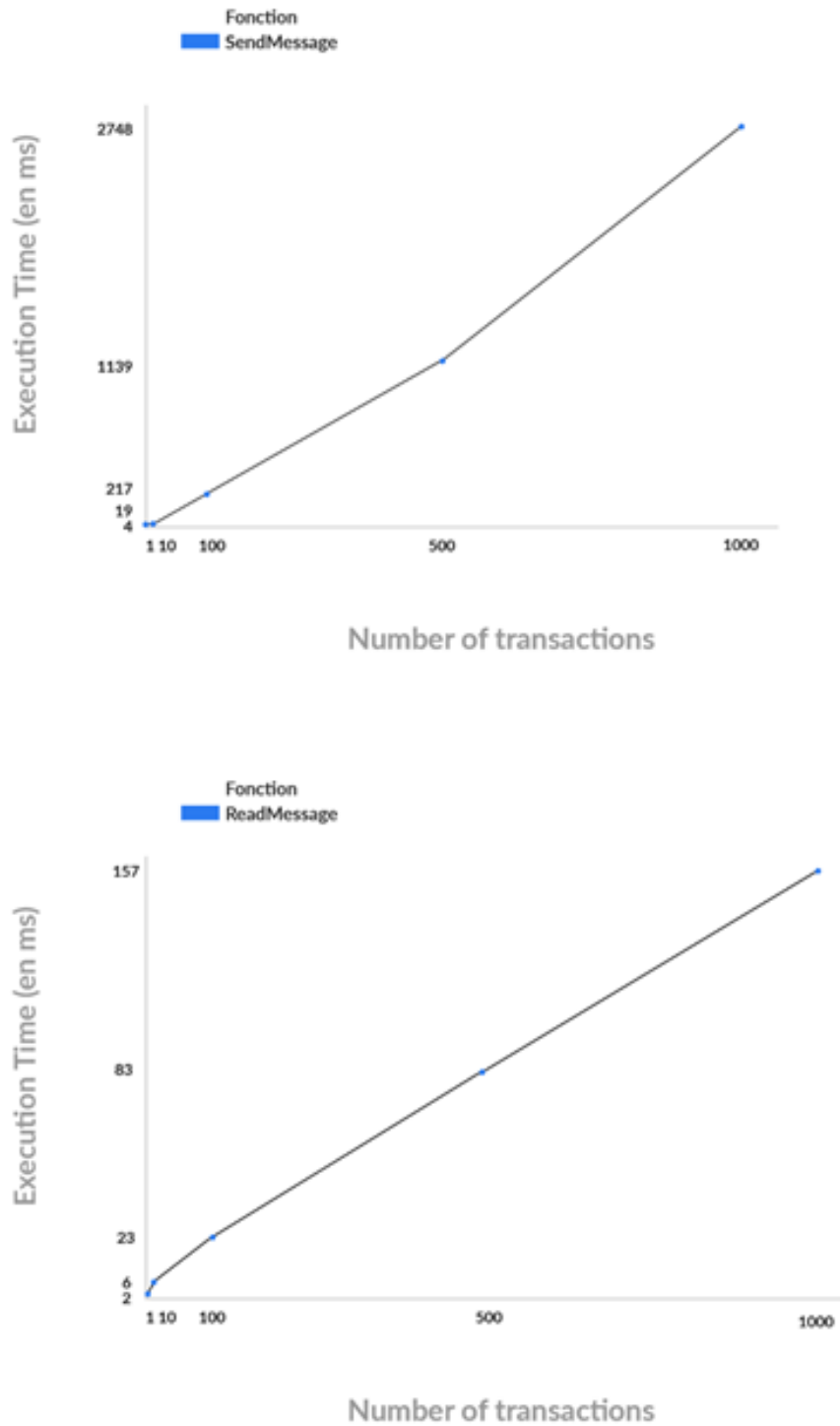


Figure S9: Temps d'exécution des fonctions les plus demandées de PSEV-Communication.

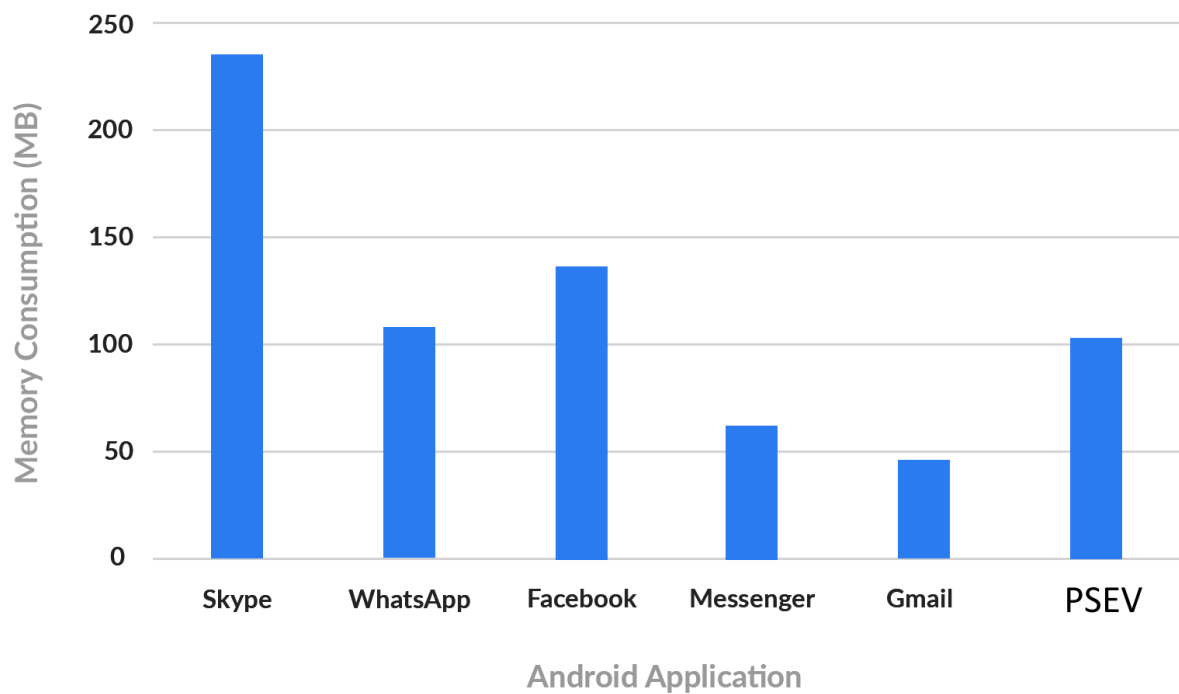


Figure S10: La comparaison de la consommation de mémoire du PSEV avec des applications mobiles commerciales.

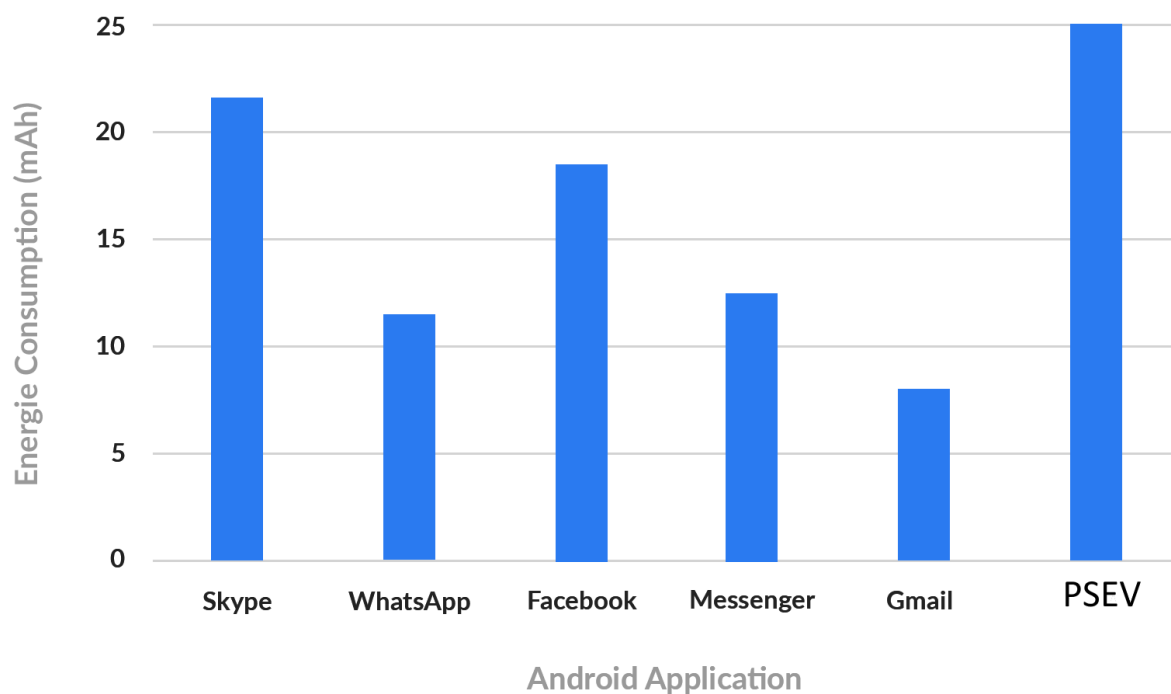


Figure S11: La comparaison de la consommation d'énergie du PSEV avec des applications mobiles commerciales.

Coût

La PSEV se compose de trois systèmes dont chacun est doté de fonctions particulières. Le système de gestion d'accès contient les fonctions "AddVehicles", "GetVehiclesByAdress", "AddParking", "GetParkingByAddress" et "GetAllParking". Le système de gestion de la communication comprend les fonctions "SendMessage" et "ReadMessage". Finalement, le système de gestion de paiement du parking englobe les fonctions "SellTicket", "GetNumberOfTicket" et "BuyTicket". La formule la plus chère est la génération et le déploiement du prototype sur la Blockchain (30,7 USD), suivie de "BuyTicket" (0,2739 USD), "SellTicket" (0,10613 USD), "AddParking" (0,09561 USD), "AddVehicles" (0,08446 USD) et "SendMessage" (0,05677 USD) tandis que les autres n'engendrent pas de coûts supplémentaires. En revanche, DISV contient seulement trois fonctions : DeployContract (0,07572 USD une seule fois), "SetMessage" (0,02486 USD -0,0456 USD) et "GetMessage" (0). Par conséquent, la comparaison a révélé que le déploiement de PSEV est plus cher car il contient plus de contrats et de fonctionnalités.

Temps d'exécution

Le temps d'exécution est un critère fondamental pour l'évaluation des systèmes de communication tels que la PSEV, compte tenu des sérieuses perturbations engendrées par les petits retards. La PSEV prend entre 2 et 57 millisecondes pour appeler une fonction quelconque, ce qui prouve qu'il s'agit d'une application en temps réel. Avec la DISV, le temps de réponse de messagerie prend entre 0 et 3s. La PSEV est également considérée comme une application en temps réel. Il convient de noter que la PSEV est capable d'envoyer 1145 messages en 3 secondes, ce qui surpasse les solutions similaires telles que DISV qui ne peut envoyer que 25 à tous les participants au sein des systèmes de transport intelligents. De plus, quand on compare les deux solutions, il faut moins de temps pour envoyer et recevoir un message dans PSEV alors que la vente et l'achat d'un billet prend plus de temps.

Consommation de la mémoire et de l'énergie

L'évaluation de la consommation de la mémoire et de l'énergie est requise compte tenu des capacités informatiques et énergétiques faibles des appareils de l'IdO. La PSEV consomme 25.43 mA, similairement aux applications commerciales telles que Facebook, Skype, Gmail, Messenger et WhatsApp. La DISV consomme une moyenne d'électricité de 23,43 mAh. Plus précisément, la PSEV requiert plus d'énergie que la DISV car elle est dotée de plus de fonctionnalités. La comparaison a révélé que les

deux solutions sont des applications en temps réel. La PSEV est plus chère et elle consomme plus d'énergie et de mémoire car il s'agit d'une solution plus complexe avec plus de fonctions à exécuter.

4. Conclusion

Les Réseaux Ad-Hoc des véhicules (VANET) ont été révolutionnés et transformés en "Internet des Véhicules" (IoV) grâce aux technologies innovantes de l'IdO. L'IoV utilise les plateformes d'informations, les systèmes de navigation des véhicules, les terminaux de paiement intelligents et la technologie de communication mobile pour établir une interaction de données en temps réel entre les véhicules et les piétons, les véhicules et les infrastructures. En étant connectés à internet, les véhicules qui font partie de l'IdO deviennent connectés aux systèmes d'information de la circulation et aux autres véhicules également. Toutefois, compte tenu de la sensibilité des données échangées et de la haute connectivité, la mise en œuvre peut mettre la confidentialité et la sécurité en péril, ce qui est susceptible d'exposer les véhicules aux attaques malicieuses. Dans cette étude, nous avons développé deux solutions avec les technologies de la Blockchain et l'IoV pour relever les défis liés à la confidentialité et la sécurité et éviter que les systèmes interconnectés soient victimes des cyberattaques et de la divulgation des informations sensibles. Notre première solution proposée est DISV qui désigne "Decentralized IoT Solution for Vehicles communication", c.à.d. Solution IdO Décentralisée pour la communication des Véhicules. DISV est basée sur la Blockchain et elle vise à améliorer la sécurité de la communication en IoV. DISV permet aux participants des réseaux de l'IoV de recevoir des messages tandis que ces messages sont ajoutés simultanément à la couche choisie de la Blockchain. La deuxième solution proposée est PSEV. Elle est basée sur la Blockchain et elle utilise l'Ethereum pour générer un système de paiement intelligent pour le parking (PSEV-Payment). Lors de la conception de cette solution, nous sommes partis avec l'hypothèse que les propriétaires des espaces de parking privé désirent louer leurs places de parking lorsqu'elles ne sont pas utilisées. Cette solution fonctionne en temps réel et elle permet une communication instantanée entre tous les participants du système de transport. De plus, PSEV et DISV permettent le partage des données et la coopération entre les participants des systèmes de transport intelligents, le fonctionnement des systèmes avancés d'assistance à la conduite (ADAS) et améliorent la sécurité et la sûreté globales du système de transport. Dans cette étude, nous avons évalué les propriétés les plus importantes de la solution, à savoir la consommation de la mémoire et de l'énergie, l'immutabilité, la confidentialité, la cohérence, l'intégrité, le temps d'exécution et le

coût. L'objet de cette évaluation est de s'assurer de la capacité de la plateforme basée sur la Blockchain à assurer une communication efficace et un paiement sécurisé avec l'IoV. Selon les résultats, DISV et PSEV se sont avérés être non seulement des solutions en temps réel, mais elles sont plus évolutives et plus rapides que les solutions de communication actuelles. Par ailleurs, l'étude a démontré que DISV et PSEV peuvent contribuer à résoudre les défis les plus critiques de la communication véhicule-à-tout (V2X) en améliorant la sécurité et l'évolutivité.

Glossary of acronyms

ACID Atomicity, Consistency, Isolation and Duration.

ADAS Advanced Driver Assistance System.

AP Application for Infrastructure.

API Authentication Path Information.

ATR Application en temps réel.

AV Application for Vehicles.

BGP Border Gateway Protocol.

BGP Multi-Agent Reinforcement Learning.

BIoV Blockchain-based IoV solutions.

CAV Connected Autonomous Vehicles.

DApp Decentralized applications.

Dapp Decentralized applications.

dApp Decentralized applications.

DoS Denial-of-Service.

DPoS Delegated Proof of Stake.

EV Electric Vehicle.

EVM Ethereum Virtual Machine.

FBA Federated Byzantine Agreement.

FSMs Finite state machines.

IdO Internet Des Objets.

ILP Interledger Protocol.

IoT Internet of Things.

IoV Internet of Vehicles.

ITS Intelligent Transport Systems.

MAC Message Authentication Code.

MBT Model-Based Testing.

OWASP Open Web Application Security Project.

PBFT Practical Byzantine Fault Tolerance.

PoAc Proof of Activity.

PoAu Proof of Authority.

PoB Proof of Burn.

PoC Proof of Capacity.

PoET Proof of Elapsed Time.

PoS Proof-of-Stake.

PoW Proof-of-Work.

RTA Real-Time Application.

SCSVS Smart Contract Security Verification Standard.

SUT System Under Test.

TA Timed Automata.

TPS Transactions per second.

UAVs Unmanned Aerial Vehicles.

V2I Vehicle-to-Infrastructure.

V2V Vehicle-to-Vehicle.

V2X Vehicle-to-Everything.

VANET Vehicle Ad-hoc Networks.

VDCS Vehicle Data Collection System.

WCF Windows Communication Foundation.

XEE XML External Entities.

Bibliography

- [1] S. Tanwar, S. Tyagi, I. Budhiraja et N. Kumar, “Tactile internet for autonomous vehicles: Latency and reliability analysis,” *IEEE Wireless Communications*, vol. 26, n^o. 4, p. 66–72, 2019.
- [2] S. Sharma et B. Kaushik, “A survey on internet of vehicles: Applications, security issues & solutions,” *Vehicular Communications*, vol. 20, p. 100182, 2019.
- [3] Y. Dai, D. Xu, S. Maharjan, G. Qiao et Y. Zhang, “Artificial intelligence empowered edge computing and caching for internet of vehicles,” *IEEE Wireless Communications*, vol. 26, n^o. 3, p. 12–18, 2019.
- [4] A. Lamssaggad, N. Benamar, A. S. Hafid et M. Msahli, “A survey on the current security landscape of intelligent transportation systems,” *IEEE Access*, vol. 9, p. 9180–9208, 2021.
- [5] K. Zheng, Q. Zheng, P. Chatzimisios, W. Xiang et Y. Zhou, “Heterogeneous vehicular networking: A survey on architecture, challenges, and solutions,” *IEEE communications surveys & tutorials*, vol. 17, n^o. 4, p. 2377–2396, 2015.
- [6] G. Karagiannis, O. Altintas, E. Ekici, G. Heijenk, B. Jarupan, K. Lin et T. Weil, “Vehicular networking: A survey and tutorial on requirements, architectures, challenges, standards and solutions,” *IEEE communications surveys & tutorials*, vol. 13, n^o. 4, p. 584–616, 2011.
- [7] “ERTICO – ITS Europe,” <https://ertico.com/>, 2021, [Online; Available].
- [8] “cityverve,” <https://www.smartsustainablecities.uk/cityverve>, 2021, [Online; Available].
- [9] M. Nofer, P. Gomber, O. Hinz et D. Schiereck, “Blockchain,” *Business & Information Systems Engineering*, vol. 59, n^o. 3, p. 183–187, 2017.
- [10] “bitcoin.org,” <https://https://www.bitcoin.org/>, 2021, [Online; Available].

BIBLIOGRAPHY

- [11] A. Reyna, C. Martín, J. Chen, E. Soler et M. Díaz, “On blockchain and its integration with iot. challenges and opportunities,” *Future generation computer systems*, vol. 88, p. 173–190, 2018.
- [12] Y. Lu, “Blockchain: A survey on functions, applications and open issues,” *Journal of Industrial Integration and Management*, vol. 3, n^o. 04, p. 1850015, 2018.
- [13] Y. Yuan et F.-Y. Wang, “Towards blockchain-based intelligent transportation systems,” dans *2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC)*. IEEE, 2016, p. 2663–2668.
- [14] M. Mettler, “Blockchain technology in healthcare: The revolution starts here,” dans *2016 IEEE 18th international conference on e-health networking, applications and services (Healthcom)*. IEEE, 2016, p. 1–3.
- [15] “Hyperledger Iroha ,” <https://www.hyperledger.org/projects/iroha>, 2021, [Online; Available].
- [16] S. A. Abeyratne et R. Monfared, “Blockchain ready manufacturing supply chain using distributed ledger,” *International Journal of Research in Engineering and Technology*, vol. 05, p. 1–10, 2016.
- [17] J. Yoo, Y. Jung, D. Shin, M. Bae et E. Jee, “Formal modeling and verification of a federated byzantine agreement algorithm for blockchain platforms,” dans *2019 IEEE International Workshop on Blockchain Oriented Software Engineering (IWBOSE)*. IEEE, 2019, p. 11–21.
- [18] Z. Li, Z. Xiao, Q. Xu, E. Sotthiwat, R. S. M. Goh et X. Liang, “Blockchain and iot data analytics for fine-grained transportation insurance,” dans *2018 IEEE 24th International Conference on Parallel and Distributed Systems (ICPADS)*. IEEE, 2018, p. 1022–1027.
- [19] A. Zhang et X. Lin, “Towards secure and privacy-preserving data sharing in e-health systems via consortium blockchain,” *Journal of medical systems*, vol. 42, n^o. 8, p. 140, 2018.
- [20] W. Wang, D. T. Hoang, Z. Xiong, D. Niyato, P. Wang, P. Hu et Y. Wen, “A survey on consensus mechanisms and mining management in blockchain networks,” *arXiv preprint arXiv:1805.02707*, p. 1–33, 2018.
- [21] Z. Li, J. Kang, R. Yu, D. Ye, Q. Deng et Y. Zhang, “Consortium blockchain for secure energy trading in industrial internet of things,” *IEEE transactions on industrial informatics*, vol. 14, n^o. 8, p. 3690–3700, 2017.

BIBLIOGRAPHY

- [22] J. Kang, Z. Xiong, D. Niyato, P. Wang, D. Ye et D. I. Kim, “Incentivizing consensus propagation in proof-of-stake based consortium blockchain networks,” *IEEE Wireless Communications Letters*, vol. 8, n^o. 1, p. 157–160, 2018.
- [23] “Hyperledger Hyperledger-burrow ,” www.hyperledger.org/projects/hyperledger-burrow, 2021, [Online; Available].
- [24] “Hyperledger Fabric ,” <https://www.hyperledger.org/projects/fabric>, 2021, [Online; Available].
- [25] “Hyperledger Indy ,” <https://www.hyperledger.org/projects/hyperledger-indy>, 2021, [Online; Available].
- [26] H. Li, R. Lu, L. Zhou, B. Yang et X. Shen, “An efficient merkle-tree-based authentication scheme for smart grid,” *IEEE Systems Journal*, vol. 8, n^o. 2, p. 655–663, 2013.
- [27] M. Andoni, V. Robu, D. Flynn, S. Abram, D. Geach, D. Jenkins, P. McCallum et A. Peacock, “Blockchain technology in the energy sector: A systematic review of challenges and opportunities,” *Renewable and Sustainable Energy Reviews*, vol. 100, p. 143–174, 2019.
- [28] “Hyperledger Quilt,” <https://www.hyperledger.org/projects/quilt>, 2021, [Online; Available].
- [29] “Hyperledger Sawtooth ,” <https://www.hyperledger.org/projects/sawtooth>, 2021, [Online; Available].
- [30] “Interledger Protocol (ILP),” <https://interledger.org/rfcs/0003-interledger-protocol>, 2021, [Online; Available].
- [31] D. Mingxiao, M. Xiaofeng, Z. Zhe, W. Xiangwei et C. Qijun, “A review on consensus algorithm of blockchain,” dans *2017 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*. IEEE, 2017, p. 2567–2572.
- [32] “A (Short) Guide to Blockchain Consensus Protocols,” <https://www.coindesk.com/short-guide-blockchain-consensus-protocols/>, 2021, [Online; Available].
- [33] S. Gupta et M. Sadoghi, “Blockchain transaction processing.” 2019.

- [34] H. Hou, “The application of blockchain technology in e-government in china,” dans *2017 26th International Conference on Computer Communication and Networks (ICCCN)*. IEEE, 2017, p. 1–4.
- [35] S. Nakamoto et A. Bitcoin, “A peer-to-peer electronic cash system,” *Bitcoin*.–URL: <https://bitcoin.org/bitcoin.pdf>, vol. 4, 2008.
- [36] S. Haber et W. S. Stornetta, “How to time-stamp a digital document,” *J. Cryptology*, vol. 3, p. 99–111, 1991.
- [37] N. Szabo, “Bit gold ,” <https://unenumerated.blogspot.com/2005/12/bit-gold.html>, 2008, [Online; Available].
- [38] A. Luntovskyy et D. Guetter, “Cryptographic technology blockchain and its applications,” dans *The International Conference on Information and Telecommunication Technologies and Radio Electronics*. Springer, 2018, p. 14–33.
- [39] A. Gervais, G. O. Karame, K. Wüst, V. Glykantzis, H. Ritzdorf et S. Capkun, “On the security and performance of proof of work blockchains,” dans *Proceedings of the 2016 ACM SIGSAC conference on computer and communications security*, 2016, p. 3–16.
- [40] L. Li, J. Liu, L. Cheng, S. Qiu, W. Wang, X. Zhang et Z. Zhang, “Creditcoin: A privacy-preserving blockchain-based incentive announcement network for communications of smart vehicles,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 19, n^o. 7, p. 2204–2220, 2018.
- [41] F. Gao, L. Zhu, M. Shen, K. Sharif, Z. Wan et K. Ren, “A blockchain-based privacy-preserving payment mechanism for vehicle-to-grid networks,” *IEEE network*, vol. 32, n^o. 6, p. 184–192, 2018.
- [42] H. Liu, Y. Zhang et T. Yang, “Blockchain-enabled security in electric vehicles cloud and edge computing,” *IEEE Network*, vol. 32, n^o. 3, p. 78–83, 2018.
- [43] E. Erdin, M. Cebe, K. Akkaya, S. Solak, E. Bulut et S. Uluagac, “Building a private bitcoin-based payment network among electric vehicles and charging stations,” dans *2018 IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications*

BIBLIOGRAPHY

- (*GreenCom*) and *IEEE Cyber, Physical and Social Computing (CPSCoM)* and *IEEE Smart Data (SmartData)*. IEEE, 2018, p. 1609–1615.
- [44] A. Lei, H. Cruickshank, Y. Cao, P. Asuquo, C. P. A. Ogah et Z. Sun, “Blockchain-based dynamic key management for heterogeneous intelligent transportation systems,” *IEEE Internet of Things Journal*, vol. 4, n^o. 6, p. 1832–1843, 2017.
- [45] M. Liu, Y. Teng, F. R. Yu, V. C. Leung et M. Song, “Deep reinforcement learning based performance optimization in blockchain-enabled internet of vehicle,” dans *ICC 2019-2019 IEEE International Conference on Communications (ICC)*. IEEE, 2019, p. 1–6.
- [46] S. Jeong, N.-N. Dao, Y. Lee, C. Lee et S. Cho, “Blockchain based billing system for electric vehicle and charging station,” dans *2018 Tenth International Conference on Ubiquitous and Future Networks (ICUFN)*. IEEE, 2018, p. 308–310.
- [47] W. Hu, W. Yao, Y. Hu et H. Li, “Collaborative optimization of distributed scheduling based on blockchain consensus mechanism considering battery-swap stations of electric vehicles,” *IEEE Access*, vol. 7, p. 137 959–137 967, 2019.
- [48] J. Innerbichler et V. Damjanovic-Behrendt, “Federated byzantine agreement to ensure trustworthiness of digital manufacturing platforms,” dans *Proceedings of the 1st Workshop on Cryptocurrencies and Blockchains for Distributed Systems*, 2018, p. 111–116.
- [49] D. Mazieres, “The stellar consensus protocol: A federated model for internet-level consensus,” *Stellar Development Foundation*, vol. 32, 2015.
- [50] “Stellar. [Online]. Available: <https://www.stellar.org/>,” 2020.
- [51] “Ripple. [Online]. Available: <https://ripple.com/>,” 2020.
- [52] “Intel is winning over blockchain critics by reimagining Bitcoin’s DNA.” www.coindesk.com/intel-winning-blockchain-critics-reimagining-bitcoins-dnal, 2016, [Online; Available].
- [53] D. D. Baby, K. C Sivarama, S. P. Cimryn et V. N, “Tracking and monitoring of vehicles and a stable and secure tolltax payment methodology based on blockchain enabled cryptocurrency e-wallets,” *International Journal of Engineering and Advanced Technology (IJEAT)*, vol. 8, p. 685–690, 2019.

BIBLIOGRAPHY

- [54] Z. Zhou, B. Wang, M. Dong et K. Ota, "Secure and efficient vehicle-to-grid energy trading in cyber physical systems: Integration of blockchain and edge computing," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 50, n^o. 1, p. 43–57, 2020.
- [55] Y. Wang, Z. Su, Q. Xu et N. Zhang, "Contract based energy blockchain for secure electric vehicles charging in smart community," dans *2018 IEEE 16th Intl Conf on Dependable, Autonomic and Secure Computing, 16th Intl Conf on Pervasive Intelligence and Computing, 4th Intl Conf on Big Data Intelligence and Computing and Cyber Science and Technology Congress(DASC/PiCom/DataCom/CyberSciTech)*, 2018, p. 323–327.
- [56] "Ethereum. Blockchain App Platform. [Online]. Available: <https://ethereum.org/>," 2020.
- [57] "Hyperledger," <https://www.ibm.com/blockchain/hyperledger>, 2021, [Online; Available].
- [58] "monax. [Online]. Available: <https://monax.io/>," 2021.
- [59] "Hyperledger Caliper," <https://www.hyperledger.org/projects/caliper>, 2021, [Online; Available].
- [60] "Hyperledger Cello," <https://www.hyperledger.org/projects/cello>, 2021, [Online; Available].
- [61] "Hyperledger Composer project," <https://www.hyperledger.org/projects/composer>, 2021, [Online; Available].
- [62] "Hyperledger Explore," <https://www.hyperledger.org/projects/explorer>, 2021, [Online; Available].
- [63] D. Boughaci et O. Boughaci, "A comparative study of three blockchain emerging technologies: Bitcoin, ethereum and hyperledger," dans *International Conference on Computing*. Springer, 2019, p. 3–7.
- [64] L. Mendiboure, M. A. Chalouf et F. Krief, "Survey on blockchain-based applications in internet of vehicles," *Computers & Electrical Engineering*, vol. 84, p. 106646, 2020.
- [65] L. Atzori, A. Iera, and G. Morabito, "The internet of things: A survey," *Comput. Netw.*, vol. 54, n^o. 15, p. 2787–2805, 2010.
- [66] F. Yang, S. Wang, J. Li, Z. Liu and Q. Sun, "The internet of things: A survey," *China Communications*, vol. 11(10), p. 1–15, 2014.

BIBLIOGRAPHY

- [67] “Delhi: 18,000 autos install GPS in one month, deadline extended to February 28. [Online]. Available: <https://www.hindustantimes.com/delhi-news/delhi-18-000-autos-install-gps-in-one-month-deadline-extended-to-february-28/story-xllsXOhh94aKOxaO5wwErJ.html>,” 2019.
- [68] “blockchain bitcoin ethereum vehicle vehicles transport transportation,” <https://www.scopus.com/results/results.uri?sort=plf-f&src=s&st1=%22blockchain%22+OR+%22bitcoin%22+OR+%22ethereum%22&st2=%22vehicle%22+OR+%22cars%22+OR+%22vehicles%22+OR+%22transport%7c%22+OR+%22transportation%22+OR+%22car%22+OR+%22driver%22+OR+%22Vehicular%22&sid=57d63d866eb16cbeda2f25ddbeaa3d5b&sot=b&sdt=b&sl=165&s=%28TITLE%28%22blockchain%22+OR+%22bitcoin%22+OR+%22ethereum%22%29+AND+TITLE%28%22vehicle%22+OR+%22cars%22+OR+%22vehicles%22+OR+%22transport%7c%22+OR+%22transportation%22+OR+%22car%22+OR+%22driver%22+OR+%22Vehicular%22%29%29&origin=savedSearchNewOnly&txGid=75da2dfd8374fc462ae10532ea94cae>, 2021, [Online; Available].
- [69] P. K. Sharma, S. Y. Moon et J. H. Park, “Block-vn: A distributed blockchain based vehicular network architecture in smart city.” *Journal of information processing systems*, vol. 13, n^o. 1, 2017.
- [70] F. Casino, T. K. Dasaklis et C. Patsakis, “A systematic literature review of blockchain-based applications: current status, classification and open issues,” *Telematics and Informatics*, vol. 36, p. 55–81, 2019.
- [71] V. Astarita, V. P. Giofrè, G. Mirabelli et V. Solina, “A review of blockchain-based systems in transportation,” *Information*, vol. 11, n^o. 1, p. 21, 2020.
- [72] A. Ometov, Y. Bardinova, A. Afanasyeva, P. Masek, K. Zhidanov, S. Vanurin, M. Sayfullin, V. Shubina, M. Komarov et S. Bezzateev, “An overview on blockchain for smartphones: State-of-the-art, consensus, implementation, challenges and future trends,” *IEEE Access*, vol. 8, p. 103 994–104 015, 2020.

- [73] J. Huang, K. Lei, M. Du, H. Zhao, H. Liu, J. Liu et Z. Qi, "Survey on blockchain incentive mechanism," dans *International Conference of Pioneering Computer Scientists, Engineers and Educators*. Springer, 2019, p. 386–395.
- [74] M. Pournader, Y. Shi, S. Seuring et S. L. Koh, "Blockchain applications in supply chains, transport and logistics: a systematic review of the literature," *International Journal of Production Research*, vol. 58, n^o. 7, p. 2063–2081, 2020.
- [75] M. B. Mollah, J. Zhao, D. Niyato, Y. L. Guan, C. Yuen, S. Sun, K.-Y. Lam et L. H. Koh, "Blockchain for the internet of vehicles towards intelligent transportation systems: A survey," *IEEE Internet of Things Journal*, 2020.
- [76] P. Sharma, R. Jindal et M. D. Borah, "Blockchain technology for cloud storage: A systematic literature review," *ACM Computing Surveys (CSUR)*, vol. 53, n^o. 4, p. 1–32, 2020.
- [77] S. Smetanin, A. Ometov, M. Komarov, P. Masek et Y. Koucheryavy, "Blockchain evaluation approaches: State-of-the-art and future perspective," *Sensors*, vol. 20, n^o. 12, p. 3358, 2020.
- [78] U. Bodkhe, S. Tanwar, K. Parekh, P. Khanpara, S. Tyagi, N. Kumar et M. Alazab, "Blockchain for industry 4.0: A comprehensive review," *IEEE Access*, vol. 8, p. 79 764–79 800, 2020.
- [79] Z. Yang, K. Yang, L. Lei, K. Zheng et V. C. Leung, "Blockchain-based decentralized trust management in vehicular networks," *IEEE Internet of Things Journal*, vol. 6, n^o. 2, p. 1495–1505, 2018.
- [80] Z. Yang, K. Zheng, K. Yang et V. C. Leung, "A blockchain-based reputation system for data credibility assessment in vehicular networks," dans *2017 IEEE 28th annual international symposium on personal, indoor, and mobile radio communications (PIMRC)*. IEEE, 2017, p. 1–5.
- [81] N. Malik, P. Nanda, A. Arora, X. He et D. Puthal, "Blockchain based secured identity authentication and expeditious revocation framework for vehicular networks," dans *2018 17th IEEE International Conference On Trust, Security And Privacy In Computing And Communications/12th IEEE International Conference On Big Data Science And Engineering (TrustCom/BigDataSE)*. IEEE, 2018, p. 674–679.

- [82] M. Labrador et W. Hou, “Implementing blockchain technology in the internet of vehicle (iov),” dans *2019 International Conference on Intelligent Computing and its Emerging Applications (ICEA)*. IEEE, 2019, p. 5–10.
- [83] T. Reimers, F. Leber et U. Lechner, “Integration of blockchain and internet of things in a car supply chain,” dans *2019 IEEE International Conference on Decentralized Applications and Infrastructures (DAPPCON)*. IEEE, 2019, p. 146–151.
- [84] A. Mostafa, “Vanet blockchain: A general framework for detecting malicious vehicles,” *J. Commun*, vol. 14, n^o. 5, p. 356–362, 2019.
- [85] G. Rathee, A. Sharma, R. Iqbal, M. Aloqaily, N. Jaglan et R. Kumar, “A blockchain framework for securing connected and autonomous vehicles,” *Sensors*, vol. 19, n^o. 14, p. 3165, 2019.
- [86] J. Noh, S. Jeon et S. Cho, “Distributed blockchain-based message authentication scheme for connected vehicles,” *Electronics*, vol. 9, n^o. 1, p. 74, 2020.
- [87] S. Nadeem, M. Rizwan, F. Ahmad et J. Manzoor, “Securing cognitive radio vehicular ad hoc network with fog node based distributed blockchain cloud architecture,” *INTERNATIONAL JOURNAL OF ADVANCED COMPUTER SCIENCE AND APPLICATIONS*, vol. 10, n^o. 1, p. 288–295, 2019.
- [88] Q. Hu, S. Fong, P. Qin, J. Guo, Y. Zhang, D. Xu, Y. Chen et J. Yen, “Intelligent car parking system based on blockchain processing reengineering,” dans *International Conference on e-Business Engineering*. Springer, 2019, p. 265–273.
- [89] O.-B. Kwame, Q. Xia, E. B. Sifah, S. Amofa, K. N. Acheampong, J. Gao, R. Chen, H. Xia, J. C. Gee, X. Du *et al.*, “V-chain: A blockchain-based car lease platform,” dans *2018 IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData)*. IEEE, 2018, p. 1317–1325.
- [90] G. M. Gandhi *et al.*, “Artificial intelligence integrated blockchain for training autonomous cars,” dans *2019 Fifth International Conference on Science Technology Engineering and Mathematics (ICONSTEM)*, vol. 1. IEEE, 2019, p. 157–161.

- [91] M. Z. Masoud, Y. Jaradat, I. Jannoud et D. Zaidan, “Carchain: A novel public blockchain-based used motor vehicle history reporting system,” dans *2019 IEEE Jordan International Joint Conference on Electrical Engineering and Information Technology (JEEIT)*. IEEE, 2019, p. 683–688.
- [92] Y. Zhu, F. Du, B. Wu et Z. Duan, “A sharing platform of emergency cars based on blockchain environment,” dans *International Conference on Frontier Computing*. Springer, 2019, p. 199–209.
- [93] Y. Dai, D. Xu, K. Zhang, S. Maharjan et Y. Zhang, “Deep reinforcement learning and permissioned blockchain for content caching in vehicular edge computing and networks,” *IEEE Transactions on Vehicular Technology*, vol. 69, n^o. 4, p. 4312–4324, 2020.
- [94] Y. Song, R. Yu, Y. Fu, L. Zhou et A. Boukerche, “Multi-vehicle cooperative positioning correction framework based on vehicular blockchain,” dans *Proceedings of the 9th ACM Symposium on Design and Analysis of Intelligent Vehicular Networks and Applications*, 2019, p. 23–29.
- [95] X. Zhang et D. Wang, “Adaptive traffic signal control mechanism for intelligent transportation based on a consortium blockchain,” *IEEE Access*, vol. 7, p. 97 281–97 295, 2019.
- [96] X. Huang, C. Xu, P. Wang et H. Liu, “Lnsc: A security model for electric vehicle and charging pile management based on blockchain ecosystem,” *IEEE Access*, vol. 6, p. 13 565–13 574, 2018.
- [97] M. Li, L. Zhu et X. Lin, “Efficient and privacy-preserving carpooling using blockchain-assisted vehicular fog computing,” *IEEE Internet of Things Journal*, vol. 6, n^o. 3, p. 4573–4584, 2018.
- [98] H. Khelifi, S. Luo, B. Nour, H. Mounsla et S. H. Ahmed, “Reputation-based blockchain for secure ndn caching in vehicular networks,” dans *2018 IEEE Conference on Standards for Communications and Networking (CSCN)*. IEEE, 2018, p. 1–6.
- [99] D. Zhang, F. R. Yu et R. Yang, “Blockchain-based distributed software-defined vehicular networks: A dueling deep Q -learning approach,” *IEEE Transactions on Cognitive Communications and Networking*, vol. 5, n^o. 4, p. 1086–1100, 2019.

BIBLIOGRAPHY

- [100] Y. Yahiatene et A. Rachedi, “Towards a blockchain and software-defined vehicular networks approaches to secure vehicular social network,” dans *2018 IEEE Conference on Standards for Communications and Networking (CSCN)*. IEEE, 2018, p. 1–7.
- [101] L.-A. Hirtan et C. Dobre, “Blockchain privacy-preservation in intelligent transportation systems,” dans *2018 IEEE International Conference on Computational Science and Engineering (CSE)*. IEEE, 2018, p. 177–184.
- [102] J. Kang, Z. Xiong, D. Niyato, D. Ye, D. I. Kim et J. Zhao, “Toward secure blockchain-enabled internet of vehicles: Optimizing consensus management using reputation and contract theory,” *IEEE Transactions on Vehicular Technology*, vol. 68, n^o. 3, p. 2906–2920, 2019.
- [103] G. Baldini, J. L. Hernández-Ramos, G. Steri et S. N. Matheu, “Zone keys trust management in vehicular networks based on blockchain,” dans *2019 Global IoT Summit (GIoTS)*. IEEE, 2019, p. 1–6.
- [104] I. J. Jensen, D. F. Selvaraj et P. Ranganathan, “Blockchain technology for networked swarms of unmanned aerial vehicles (uavs),” dans *2019 IEEE 20th International Symposium on “A World of Wireless, Mobile and Multimedia Networks”(WoWMoM)*. IEEE, 2019, p. 1–7.
- [105] J. Qiu, D. Grace, G. Ding, J. Yao et Q. Wu, “Blockchain-based secure spectrum trading for unmanned-aerial-vehicle-assisted cellular networks: An operator’s perspective,” *IEEE Internet of Things Journal*, vol. 7, n^o. 1, p. 451–466, 2019.
- [106] Y. Yao, X. Chang, J. Mišić, V. B. Mišić et L. Li, “Bla: Blockchain-assisted lightweight anonymous authentication for distributed vehicular fog services,” *IEEE Internet of Things Journal*, vol. 6, n^o. 2, p. 3775–3784, 2019.
- [107] H. Chai, S. Leng, K. Zhang et S. Mao, “Proof-of-reputation based-consortium blockchain for trust resource sharing in internet of vehicles,” *IEEE Access*, vol. 7, p. 175 744–175 757, 2019.
- [108] F. Kandah, B. Huber, A. Skjellum et A. Altarawneh, “A blockchain-based trust management approach for connected autonomous vehicles in smart cities,” dans *2019 IEEE 9th Annual Computing and Communication Workshop and Conference (CCWC)*. IEEE, 2019, p. 0544–0549.

- [109] X. Wang, P. Zeng, N. Patterson, F. Jiang et R. Doss, “An improved authentication scheme for internet of vehicles based on blockchain technology,” *IEEE access*, vol. 7, p. 45 061–45 072, 2019.
- [110] J. Kang, Z. Xiong, D. Niyato et D. I. Kim, “Incentivizing secure block verification by contract theory in blockchain-enabled vehicular networks,” dans *ICC 2019-2019 IEEE International Conference on Communications (ICC)*. IEEE, 2019, p. 1–7.
- [111] M. Li, J. Weng, A. Yang, J.-N. Liu et X. Lin, “Toward blockchain-based fair and anonymous ad dissemination in vehicular networks,” *IEEE Transactions on Vehicular Technology*, vol. 68, n^o. 11, p. 11 248–11 259, 2019.
- [112] Y. Mu, F. Rezaeibagha et K. Huang, “Policy-driven blockchain and its applications for transport systems,” *IEEE Transactions on Services Computing*, vol. 13, n^o. 2, p. 230–240, 2019.
- [113] Q. Feng, D. He, S. Zeadally et K. Liang, “Bpas: Blockchain-assisted privacy-preserving authentication system for vehicular ad hoc networks,” *IEEE Transactions on Industrial Informatics*, vol. 16, n^o. 6, p. 4146–4155, 2019.
- [114] X. Li, Y. Wang, P. Vijayakumar, D. He, N. Kumar et J. Ma, “Blockchain-based mutual-healing group key distribution scheme in unmanned aerial vehicles ad-hoc network,” *IEEE Transactions on Vehicular Technology*, vol. 68, n^o. 11, p. 11 309–11 322, 2019.
- [115] B. Chen, L. Wu, H. Wang, L. Zhou et D. He, “A blockchain-based searchable public-key encryption with forward and backward privacy for cloud-assisted vehicular social networks,” *IEEE Transactions on Vehicular Technology*, 2019.
- [116] M. Shen, J. Zhang, L. Zhu, K. Xu et X. Tang, “Secure svm training over vertically-partitioned datasets using consortium blockchain for vehicular social networks,” *IEEE Transactions on Vehicular Technology*, 2019.
- [117] T. A. Butt, R. Iqbal, K. Salah, M. Aloqaily et Y. Jararweh, “Privacy management in social internet of vehicles: review, challenges and blockchain based solutions,” *IEEE Access*, vol. 7, p. 79 694–79 713, 2019.
- [118] S. Guo, X. Hu, Z. Zhou, X. Wang, F. Qi et L. Gao, “Trust access authentication in vehicular network based on blockchain,” *China Communications*, vol. 16, n^o. 6, p. 18–30, 2019.

BIBLIOGRAPHY

- [119] Q. Wang, T. Ji, Y. Guo, L. Yu, X. Chen et P. Li, “Trafficchain: A blockchain-based secure and privacy-preserving traffic map,” *IEEE Access*, vol. 8, p. 60 598–60 612, 2020.
- [120] C. Kaiser, M. Steger, A. Dorri, A. Festl, A. Stocker, M. Fellmann et S. Kanhere, “Towards a privacy-preserving way of vehicle data sharing—a case for blockchain technology?” dans *International Forum on Advanced Microsystems for Automotive Applications*. Springer, 2018, p. 111–122.
- [121] L. Zavolokina, N. Zani et G. Schwabe, “Why should i trust a blockchain platform? designing for trust in the digital car dossier,” dans *International Conference on Design Science Research in Information Systems and Technology*. Springer, 2019, p. 269–283.
- [122] I. García-Magariño, R. Lacuesta, M. Rajarajan et J. Lloret, “Security in networks of unmanned aerial vehicles for surveillance with an agent-based approach inspired by the principles of blockchain,” *Ad Hoc Networks*, vol. 86, p. 72–82, 2019.
- [123] R. Chaudhary, A. Jindal, G. S. Auja, S. Aggarwal, N. Kumar et K.-K. R. Choo, “Best: Blockchain-based secure energy trading in sdn-enabled intelligent transportation system,” *Computers & Security*, vol. 85, p. 288–299, 2019.
- [124] I. Ali, M. Gervais, E. Ahene et F. Li, “A blockchain-based certificateless public key signature scheme for vehicle-to-infrastructure communication in vanets,” *Journal of Systems Architecture*, vol. 99, p. 101636, 2019.
- [125] A. Busygin, A. Konoplev, M. Kalinin et D. Zegzhda, “Floating genesis block enhancement for blockchain based routing between connected vehicles and software-defined vanet security services,” dans *Proceedings of the 11th International Conference on Security of Information and Networks*, 2018, p. 1–2.
- [126] A. Imeri, C. Feltus, D. Khadraoui, N. Agoulmine et D. Nicolas, “Solving the trust issues in the process of transportation of dangerous goods by using blockchain technology,” dans *Proceedings of the 11th International Conference on Security of Information and Networks*, 2018, p. 1–2.
- [127] V. Valaštín, K. Košťál, R. Bencel et I. Kotuliak, “Blockchain based car-sharing platform,” dans *2019 International Symposium ELMAR*. IEEE, 2019, p. 5–8.

BIBLIOGRAPHY

- [128] B. Yin, L. Mei, Z. Jiang et K. Wang, “Joint cloud collaboration mechanism between vehicle clouds based on blockchain,” dans *2019 IEEE International Conference on Service-Oriented System Engineering (SOSE)*. IEEE, 2019, p. 227–2275.
- [129] J. Zhang, X. Huang, W. Ni, M. Wu et R. Yu, “Veschain: Leveraging consortium blockchain for secure and efficient vehicular crowdsensing,” dans *2019 Chinese Control Conference (CCC)*. IEEE, 2019, p. 6339–6344.
- [130] C. Chen, T. Xiao, T. Qiu, N. Lv et Q. Pei, “Smart-contract-based economical platooning in blockchain-enabled urban internet of vehicles,” *IEEE Transactions on Industrial Informatics*, vol. 16, n^o. 6, p. 4122–4133, 2019.
- [131] P. G. Saranti, D. Chondrogianni et S. Karatzas, “Autonomous vehicles and blockchain technology are shaping the future of transportation,” dans *The 4th conference on sustainable urban mobility*. Springer, 2018, p. 797–803.
- [132] S. A. Bagloee, M. Tavana, G. Withers, M. Patriksson et M. Asadi, “Tradable mobility permit with bitcoin and ethereum—a blockchain application in transportation,” *Internet of Things*, vol. 8, p. 100103, 2019.
- [133] S.-Y. Cho, N. Chen et X. Hua, “Developing a vehicle networking platform based on blockchain technology,” dans *International Conference on Blockchain*. Springer, 2019, p. 186–201.
- [134] C. Guo, X. Huang, C. Zhu, X. Wang et X. Cao, “Distributed electric vehicle control model based on blockchain,” dans *IOP Conference Series: Materials Science and Engineering*, vol. 486, n^o. 1. IOP Publishing, 2019, p. 012046.
- [135] S. S. Ramachandran, A. Veeraraghavan, U. Karni et K. Sivaraman, “Development of flexible autonomous car system using machine learning and blockchain,” dans *International Symposium of Information and Internet Technology*. Springer, 2018, p. 63–72.
- [136] M. N. Postorino et G. M. Sarné, “A preliminary study for an agent blockchain-based framework supporting dynamic car-pooling.” dans *WOA*, 2019, p. 65–70.
- [137] P. REN, J. XU, Y. WANG et X. MA, “Research and implementation of car rental alliance based on blockchain and internet of vehicles,” *Journal of Applied Sciences*, n^o. 6, p. 10, 2019.

BIBLIOGRAPHY

- [138] Z. Abubaker, M. U. Gurmani, T. Sultana, S. Rizwan, M. Azeem, M. Z. Iftikhar et N. Javaid, “Decentralized mechanism for hiring the smart autonomous vehicles using blockchain,” dans *International Conference on Broadband and Wireless Computing, Communication and Applications*. Springer, 2019, p. 733–746.
- [139] Y. Fu, F. R. Yu, C. Li, T. H. Luan et Y. Zhang, “Vehicular blockchain-based collective learning for connected and autonomous vehicles,” *IEEE Wireless Communications*, vol. 27, n^o. 2, p. 197–203, 2020.
- [140] V. Davydov et S. Bezzateev, “Accident detection in internet of vehicles using blockchain technology,” dans *2020 International Conference on Information Networking (ICOIN)*. IEEE, 2020, p. 766–771.
- [141] Y. Song, Y. Fu, F. R. Yu et L. Zhou, “Blockchain-enabled internet of vehicles with cooperative positioning: A deep neural network approach,” *IEEE Internet of Things Journal*, vol. 7, n^o. 4, p. 3485–3498, 2020.
- [142] S. Narbayeva, T. Bakibayev, K. Abeshev, I. Makarova, K. Shubenkova et A. Pashkevich, “Blockchain technology on the way of autonomous vehicles development,” *Transportation Research Procedia*, vol. 44, p. 168–175, 2020.
- [143] Z. Ying, M. Ma et L. Yi, “Bavpm: Practical autonomous vehicle platoon management supported by blockchain technique,” dans *2019 4th International Conference on Intelligent Transportation Engineering (ICITE)*. IEEE, 2019, p. 256–260.
- [144] A. Islam et S. Y. Shin, “A blockchain-based secure healthcare scheme with the assistance of unmanned aerial vehicle in internet of things,” *Computers & Electrical Engineering*, vol. 84, p. 106627, 2020.
- [145] S. Iqbal, A. W. Malik, A. U. Rahman et R. M. Noor, “Blockchain-based reputation management for task offloading in micro-level vehicular fog network,” *IEEE Access*, vol. 8, p. 52 968–52 980, 2020.
- [146] X. Huang, D. Ye, R. Yu et L. Shu, “Securing parked vehicle assisted fog computing with blockchain and optimal smart contract design,” *IEEE/CAA Journal of Automatica Sinica*, vol. 7, n^o. 2, p. 426–441, 2020.

- [147] X. Chen et X. Zhang, “Secure electricity trading and incentive contract model for electric vehicle based on energy blockchain,” *IEEE Access*, vol. 7, p. 178 763–178 778, 2019.
- [148] Z. Zhou, L. Tan et G. Xu, “Blockchain and edge computing based vehicle-to-grid energy trading in energy internet,” dans *2018 2nd IEEE Conference on Energy Internet and Energy System Integration (EI2)*, 2018, p. 1–5.
- [149] Z. Zhou, B. Wang, Y. Guo et Y. Zhang, “Blockchain and computational intelligence inspired incentive-compatible demand response in internet of electric vehicles,” *IEEE Transactions on Emerging Topics in Computational Intelligence*, vol. 3, n^o. 3, p. 205–216, 2019.
- [150] M. Pustišek, A. Kos et U. Sedlar, “Blockchain based autonomous selection of electric vehicle charging station,” dans *2016 International Conference on Identification, Information and Knowledge in the Internet of Things (IIKI)*, 2016, p. 217–222.
- [151] C. Gorenflo, L. Golab et S. Keshav, “Mitigating trust issues in electric vehicle charging using a blockchain,” dans *Proceedings of the Tenth ACM International Conference on Future Energy Systems*, 2019, p. 160–164.
- [152] H. Sun, S. Hua, E. Zhou, B. Pi, J. Sun et K. Yamashita, “Using ethereum blockchain in internet of things: A solution for electric vehicle battery refueling,” dans *Blockchain – ICBC 2018*, S. Chen, H. Wang et L.-J. Zhang, édit. Cham: Springer International Publishing, 2018, p. 3–17.
- [153] C. Liu, K. K. Chai, E. T. Lau et Y. Chen, “Blockchain based energy trading model for electric vehicle charging schemes,” dans *Smart Grid and Innovative Frontiers in Telecommunications*, P. H. J. Chong, B.-C. Seet, M. Chai et S. U. Rehman, édit. Cham: Springer International Publishing, 2018, p. 64–72.
- [154] S. Thakur et J. G. Breslin, “Electric vehicle charging queue management with blockchain,” dans *Internet of Vehicles. Technologies and Services Towards Smart City*, A. M. Skulimowski, Z. Sheng, S. Khemiri-Kallel, C. Cérin et C.-H. Hsu, édit. Cham: Springer International Publishing, 2018, p. 249–264.
- [155] A. R. Pedrosa et G. Pau, “Chargeltup: On blockchain-based technologies for autonomous vehicles,” dans *Proceedings of the 1st Workshop on Cryptocurrencies and Blockchains for Distributed Systems*, 2018, p. 87–92.

BIBLIOGRAPHY

- [156] F. C. Silva, M. A. Ahmed, J. M. Martínez et Y.-C. Kim, “Design and implementation of a blockchain-based energy trading platform for electric vehicles in smart campus parking lots,” *Energies*, vol. 12, n^o. 24, p. 4814, 2019.
- [157] Z. Fu, P. Dong et Y. Ju, “An intelligent electric vehicle charging system for new energy companies based on consortium blockchain,” *Journal of Cleaner Production*, p. 121219, 2020.
- [158] Y. Li et B. Hu, “An iterative two-layer optimization charging and discharging trading scheme for electric vehicle using consortium blockchain,” *IEEE Transactions on Smart Grid*, vol. 11, n^o. 3, p. 2627–2637, 2019.
- [159] M. M. Islam, M. Shahjalal, M. K. Hasan et Y. M. Jang, “Blockchain-based energy transaction model for electric vehicles in v2g network,” dans *2020 International Conference on Artificial Intelligence in Information and Communication (ICAIIIC)*. IEEE, 2020, p. 628–630.
- [160] I. A. Umoren, S. S. Jaffary, M. Z. Shakir, K. Katzis et H. Ahmadi, “Blockchain-based energy trading in electric vehicle enabled microgrids,” *IEEE Consumer Electronics Magazine*, 2020.
- [161] C. Liu, K. K. Chai, X. Zhang, E. T. Lau et Y. Chen, “Adaptive blockchain-based electric vehicle participation scheme in smart grid platform,” *IEEE Access*, vol. 6, p. 25 657–25 665, 2018.
- [162] N. Zhao et H. Wu, “Blockchain combined with smart contract to keep safety energy trading for autonomous vehicles,” dans *2019 IEEE 89th Vehicular Technology Conference (VTC2019-Spring)*, 2019, p. 1–5.
- [163] U. Asfia, V. Kamuni, A. Sheikh, S. Wagh et D. Patel, “Energy trading of electric vehicles using blockchain and smart contracts,” dans *2019 18th European Control Conference (ECC)*, 2019, p. 3958–3963.
- [164] Z. Su, Y. Wang, Q. Xu, M. Fei, Y. Tian et N. Zhang, “A secure charging scheme for electric vehicles with smart communities in energy blockchain,” *IEEE Internet of Things Journal*, vol. 6, n^o. 3, p. 4601–4613, 2019.
- [165] H. Liu, Y. Zhang, S. Zheng et Y. Li, “Electric vehicle power trading mechanism based on blockchain and smart contract in v2g network,” *IEEE Access*, vol. 7, p. 160 546–160 558, 2019.

- [166] Y. Wang, Z. Su et N. Zhang, “Bsis: Blockchain-based secure incentive scheme for energy delivery in vehicular energy network,” *IEEE Transactions on Industrial Informatics*, vol. 15, n^o. 6, p. 3620–3631, 2019.
- [167] M. Kim, K. Park, S. Yu, J. Lee, Y. Park, S.-W. Lee et B. Chung, “A secure charging system for electric vehicles based on blockchain,” *Sensors*, vol. 19, n^o. 13, p. 3028, 2019.
- [168] S. Velliangiri, G. K. L. Kumar et P. Karthikeyan, “Unsupervised blockchain for safeguarding confidential information in vehicle assets transfer,” dans *2020 6th International Conference on Advanced Computing and Communication Systems (ICACCS)*. IEEE, 2020, p. 44–49.
- [169] M. Cebe, E. Erdin, K. Akkaya, H. Aksu et S. Uluagac, “Block4forensic: An integrated lightweight blockchain framework for forensics applications of connected vehicles,” *IEEE Communications Magazine*, vol. 56, n^o. 10, p. 50–57, 2018.
- [170] K. L. Brousmiche, T. Heno, C. Poulain, A. Dalmieres et E. B. Hamida, “Digitizing, securing and sharing vehicles life-cycle over a consortium blockchain: Lessons learned,” dans *2018 9th IFIP international conference on new technologies, mobility and security (NTMS)*. IEEE, 2018, p. 1–5.
- [171] T. Jiang, H. Fang et H. Wang, “Blockchain-based internet of vehicles: Distributed network architecture and performance analysis,” *IEEE Internet of Things Journal*, vol. 6, n^o. 3, p. 4640–4649, 2018.
- [172] K. L. Brousmiche, A. Durand, T. Heno, C. Poulain, A. Dalmieres et E. B. Hamida, “Hybrid cryptographic protocol for secure vehicle data sharing over a consortium blockchain,” dans *2018 IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCoM) and IEEE Smart Data (SmartData)*. IEEE, 2018, p. 1281–1286.
- [173] X. Zhang et X. Chen, “Data security sharing and storage based on a consortium blockchain in a vehicular ad-hoc network,” *IEEE Access*, vol. 7, p. 58 241–58 254, 2019.
- [174] K. Liu, W. Chen, Z. Zheng, Z. Li et W. Liang, “A novel debt-credit mechanism for blockchain-based data-trading in internet of vehicles,” *IEEE Internet of Things Journal*, vol. 6, n^o. 5, p. 9098–9111, 2019.

BIBLIOGRAPHY

- [175] S. Bao, Y. Cao, A. Lei, P. Asuquo, H. Cruickshank, Z. Sun et M. Huth, “Pseudonym management through blockchain: Cost-efficient privacy preservation on intelligent transportation systems,” *IEEE Access*, vol. 7, p. 80 390–80 403, 2019.
- [176] L. Sang-Oun, J. Hyunseok et B. Han, “Security assured vehicle data collection platform by blockchain: Service provider’s perspective,” dans *2019 21st International Conference on Advanced Communication Technology (ICACT)*. IEEE, 2019, p. 265–268.
- [177] F. Morano, C. Ferretti, A. Leporati, P. Napoletano et R. Schettini, “A blockchain technology for protection and probative value preservation of vehicle driver data,” dans *2019 IEEE 23rd International Symposium on Consumer Technologies (ISCT)*. IEEE, 2019, p. 167–172.
- [178] R. Sharma et S. Chakraborty, “B2vdm: Blockchain based vehicular data management,” dans *2018 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*. IEEE, 2018, p. 2337–2343.
- [179] F. Kandah, B. Huber, A. Altarawneh, S. Medury et A. Skjellum, “Blast: Blockchain-based trust management in smart cities and connected vehicles setup,” dans *2019 IEEE High Performance Extreme Computing Conference (HPEC)*. IEEE, 2019, p. 1–7.
- [180] S. Sharma, K. K. Ghanshala et S. Mohan, “Blockchain-based internet of vehicles (iov): An efficient secure ad hoc vehicular networking architecture,” dans *2019 IEEE 2nd 5G World Forum (5GWF)*. IEEE, 2019, p. 452–457.
- [181] S. Bao, A. Lei, H. Cruickshank, Z. Sun, P. Asuquo et W. Hathal, “A pseudonym certificate management scheme based on blockchain for internet of vehicles,” dans *2019 IEEE Intl Conf on Dependable, Autonomic and Secure Computing, Intl Conf on Pervasive Intelligence and Computing, Intl Conf on Cloud and Big Data Computing, Intl Conf on Cyber Science and Technology Congress (DASC/PiCom/CBDCCom/CyberSciTech)*. IEEE, 2019, p. 28–35.
- [182] C. Chen, J. Wu, H. Lin, W. Chen et Z. Zheng, “A secure and efficient blockchain-based data trading approach for internet of vehicles,” *IEEE Transactions on Vehicular Technology*, vol. 68, n^o. 9, p. 9110–9121, 2019.

BIBLIOGRAPHY

- [183] M. Baza, M. Nabil, N. Lasla, K. Fidan, M. Mahmoud et M. Abdallah, “Blockchain-based firmware update scheme tailored for autonomous vehicles,” dans *2019 IEEE Wireless Communications and Networking Conference (WCNC)*. IEEE, 2019, p. 1–7.
- [184] H. Guo, E. Meamari et C.-C. Shen, “Blockchain-inspired event recording system for autonomous vehicles,” dans *2018 1st IEEE International Conference on Hot Information-Centric Networking (HotICN)*. IEEE, 2018, p. 218–222.
- [185] M. Demir, O. Turetken et A. Ferworn, “Blockchain based transparent vehicle insurance management,” dans *2019 Sixth International Conference on Software Defined Systems (SDS)*. IEEE, 2019, p. 213–220.
- [186] M. A. Rahman, M. M. Rashid, S. J. Barnes et S. M. Abdullah, “A blockchain-based secure internet of vehicles management framework,” dans *2019 UK/China Emerging Technologies (UCET)*. IEEE, 2019, p. 1–4.
- [187] L. Zhang, M. Luo, J. Li, M. H. Au, K.-K. R. Choo, T. Chen et S. Tian, “Blockchain based secure data sharing system for internet of vehicles: A position paper,” *Vehicular Communications*, vol. 16, p. 85–93, 2019.
- [188] L. Zavolokina, F. Spychiger, C. Tessone et G. Schwabe, “Incentivizing data quality in blockchains for inter-organizational networks—learning from the digital car dossier,” 2018.
- [189] D. Holtkemper et S. Wieninger, “Company data in the blockchain: A juxtaposition of technological drivers and potential applications,” dans *2018 Portland International Conference on Management of Engineering and Technology (PICMET)*. IEEE, 2018, p. 1–7.
- [190] Q. Wang, L. Zhou, Z. Tang et G. Wang, “A consortium blockchain-based model for data sharing in internet of vehicles,” dans *International Conference on Smart City and Informatization*. Springer, 2019, p. 253–267.
- [191] K. Shi, L. Zhu, C. Zhang, L. Xu et F. Gao, “Blockchain-based multimedia sharing in vehicular social networks with privacy protection,” *Multimedia Tools and Applications*, p. 1–21, 2020.

BIBLIOGRAPHY

- [192] M. U. Javed, M. Rehman, N. Javaid, A. Aldegheishem, N. Alrajeh et M. Tahir, “Blockchain-based secure data storage for distributed vehicular networks,” *Applied Sciences*, vol. 10, n^o. 6, p. 2011, 2020.
- [193] C. Chen, C. Wang, T. Qiu, N. Lv et Q. Pei, “A secure content sharing scheme based on blockchain in vehicular named data networks,” *IEEE Transactions on Industrial Informatics*, vol. 16, n^o. 5, p. 3278–3289, 2019.
- [194] Y. Lu, X. Huang, K. Zhang, S. Maharjan et Y. Zhang, “Blockchain empowered asynchronous federated learning for secure data sharing in internet of vehicles,” *IEEE Transactions on Vehicular Technology*, vol. 69, n^o. 4, p. 4298–4311, 2020.
- [195] J. Wu, X. Cui, W. Hu, K. Gai, X. Liu, K. Zhang et K. Xu, “A new sustainable interchain design on transport layer for blockchain,” dans *International Conference on Smart Blockchain*. Springer, 2018, p. 12–21.
- [196] A. Bonadio, F. Chiti, R. Fantacci et V. Vespri, “An integrated framework for blockchain inspired fog communications and computing in internet of vehicles,” *Journal of Ambient Intelligence and Humanized Computing*, vol. 11, n^o. 2, p. 755–762, 2020.
- [197] M. Awais Hassan, U. Habiba, U. Ghani et M. Shoaib, “A secure message-passing framework for inter-vehicular communication using blockchain,” *International Journal of Distributed Sensor Networks*, vol. 15, n^o. 2, p. 1550147719829677, 2019.
- [198] L. Nkenyereye, B. Adhi Tama, M. K. Shahzad et Y.-H. Choi, “Secure and blockchain-based emergency driven message protocol for 5g enabled vehicular edge computing,” *Sensors*, vol. 20, n^o. 1, p. 154, 2020.
- [199] R. Koduri, S. Nandyala, M. Manalikandy *et al.*, “Secure vehicular communication using blockchain technology,” SAE Technical Paper, Rapport technique, 2020.
- [200] L. R. Abbade, F. M. Ribeiro, M. H. da Silva, A. F. Morais, E. S. de Morais, E. M. Lopes, A. M. Alberti et J. J. Rodrigues, “Blockchain applied to vehicular odometers,” *IEEE Network*, vol. 34, n^o. 1, p. 62–68, 2020.

BIBLIOGRAPHY

- [201] P. Singh, P. Khanna et S. Kumar, “Communication architecture for vehicular ad hoc networks, with blockchain security,” dans *2020 International Conference on Computation, Automation and Knowledge Management (ICCAKM)*. IEEE, 2020, p. 68–72.
- [202] A. Kumar, A. S. Yadav et D. S. Kushwaha, “Vchain: Efficient blockchain based vehicular communication protocol,” dans *2020 10th International Conference on Cloud Computing, Data Science & Engineering (Confluence)*. IEEE, 2020, p. 762–768.
- [203] V. Hassija, M. Zaid, G. Singh, A. Srivastava et V. Saxena, “Cryptober: A blockchain-based secure and cost-optimal car rental platform,” dans *2019 Twelfth International Conference on Contemporary Computing (IC3)*, 2019, p. 1–6.
- [204] J. Pajic, J. Rivera, K. Zhang et H.-A. Jacobsen, “Eva: Fair and auditable electric vehicle charging service using blockchain,” dans *Proceedings of the 12th ACM International Conference on Distributed and Event-based Systems*, 2018, p. 262–265.
- [205] A. Kulathunge et H. Dayarathna, “Communication framework for vehicular ad-hoc networks using blockchain: Case study of metro manila electric shuttle automation project,” dans *2019 International Research Conference on Smart Computing and Systems Engineering (SCSE)*. IEEE, 2019, p. 85–90.
- [206] B. Leiding, P. Memarmoshrefi et D. Hogrefe, “Self-managed and blockchain-based vehicular ad-hoc networks,” dans *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing: Adjunct*, 2016, p. 137–140.
- [207] A. A. Yusuf, D. K. Basuki, S. Sukaridhoto, Y. P. Pratama, F. B. Putra et H. Yulianus, “Armchain-a blockchain based sensor data communication for the vehicle as a mobile sensor network,” dans *2019 International Electronics Symposium (IES)*. IEEE, 2019, p. 539–543.
- [208] H. Chai, S. Leng, M. Zeng et H. Liang, “A hierarchical blockchain aided proactive caching scheme for internet of vehicles,” dans *ICC 2019-2019 IEEE International Conference on Communications (ICC)*. IEEE, 2019, p. 1–6.
- [209] F. Ahmad, C. A. Kerrache, F. Kurugollu et R. Hussain, “Realization of blockchain in named data networking-based internet-of-vehicles,” *IT Professional*, vol. 21, n^o. 4, p. 41–47, 2019.

BIBLIOGRAPHY

- [210] W. Hu, Y. Hu, W. Yao et H. Li, “A blockchain-based byzantine consensus algorithm for information authentication of the internet of vehicles,” *IEEE Access*, vol. 7, p. 139 703–139 711, 2019.
- [211] J. Gao, K. O.-B. O. Agyekum, E. B. Sifah, K. N. Acheampong, Q. Xia, X. Du, M. Guizani et H. Xia, “A blockchain-sdn-enabled internet of vehicles environment for fog computing and 5g networks,” *IEEE Internet of Things Journal*, vol. 7, n^o. 5, p. 4278–4291, 2019.
- [212] A. Kuzmin et E. Znak, “Blockchain-base structures for a secure and operate network of semi-autonomous unmanned aerial vehicles,” dans *2018 IEEE International Conference on Service Operations and Logistics, and Informatics (SOLI)*. IEEE, 2018, p. 32–37.
- [213] M. Singh et S. Kim, “Trust bit: Reward-based intelligent vehicle commination using blockchain paper,” dans *2018 IEEE 4th World Forum on Internet of Things (WF-IoT)*. IEEE, 2018, p. 62–67.
- [214] “The Mystery Behind Block Time. [Online].Available: <https://medium.facilelogin.com/the-mysterybehind-block-time-63351e35603a>,” 2021.
- [215] “codebox-image-augmentor. [Online].Available: https://github.com/codebox/image_augmentor,” 2021.
- [216] “Dlib C++ toolkit . [Online].Available: <http://dlib.net/>,” 2021.
- [217] “Getting Deep Into Ethereum: How Data Is Stored In Ethereum?” <https://hackernoon.com/getting-deep-into-ethereum-how-data-is-stored-in-ethereum-e3f669d96033>, 2020, [Online; Available].
- [218] J. Tretmans, “Testing concurrent systems: A formal approach,” dans *Proceedings of the 10th International Conference on Concurrency Theory*, ser. CONCUR ’99. Berlin, Heidelberg: Springer-Verlag, 1999, p. 46–65.
- [219] M. Krichen et R. Alroobaea, “A new model-based framework for testing security of iot systems in smart cities using attack trees and price timed automata,” dans *14th International Conference on Evaluation of Novel Approaches to Software Engineering - ENASE 2019*, 2019.

BIBLIOGRAPHY

- [220] B. Kordy, L. Piètre-Cambacédès et P. Schweitzer, “Dag-based attack and defense modeling: Don’t miss the forest for the attack trees,” *Computer Science Review*, vol. 13-14, p. 1 – 38, 2014. [En ligne]. Disponible: <http://www.sciencedirect.com/science/article/pii/S1574013714000100>
- [221] R. Kumar, E. Ruijters et M. Stoelinga, “Quantitative attack tree analysis via priced timed automata,” dans *Formal Modeling and Analysis of Timed Systems*, S. Sankaranarayanan et E. Vicario, édit. Cham: Springer International Publishing, 2015, p. 156–171.
- [222] “Drowsy Driving NHTSA reports. [Online]. Available: <https://www.nhtsa.gov/risky-driving/drowsy-driving>,” 2020.
- [223] M. Krichen, “A formal framework for black-box conformance testing of distributed real-time systems,” *IJCCBS*, vol. 3, n°. 1/2, p. 26–43, 2012.
- [224] M. Krichen et S. Tripakis, “Interesting properties of the conformance relation tioco,” dans *IC-TAC’06*, 2006.
- [225] H. Zhu et F. Belli, “Advancing test automation technology to meet the challenges of model-based software testing - guest editors’ introduction to the special section of the third IEEE international workshop on automation of software test (AST 2008),” *Information & Software Technology*, vol. 51, n°. 11, p. 1485–1486, 2009.
- [226] A. C. D. Neto et G. H. Travassos, “A picture from the model-based testing area: Concepts, techniques, and challenges,” *Advances in Computers*, vol. 80, p. 45–120, 2010.
- [227] M. Utting, A. Pretschner et B. Legeard, “A taxonomy of model-based testing approaches,” *Softw. Test. Verif. Reliab.*, vol. 22, n°. 5, p. 297–312, août 2012. [En ligne]. Disponible: <http://dx.doi.org/10.1002/stvr.456>
- [228] R. Alur et D. Dill, “A theory of timed automata,” *Theoretical Computer Science*, vol. 126, p. 183–235, 1994.
- [229] J. Sifakis et S. Yovine, “Compositional specification of timed systems,” dans *13th Annual Symposium on Theoretical Aspects of Computer Science, STACS’96*, ser. LNCS, vol. 1046. Springer-Verlag, 1996.

- [230] A. J. Maâlej, M. Hamza, M. Krichen et M. Jmaiel, “Automated significant load testing for WS-BPEL compositions,” dans *Sixth IEEE International Conference on Software Testing, Verification and Validation, ICST 2013 Workshops Proceedings, Luxembourg, Luxembourg, March 18-22, 2013*, 2013, p. 144–153. [En ligne]. Disponible: <https://doi.org/10.1109/ICSTW.2013.25>
- [231] A. J. Maâlej, M. Krichen et M. Jmaiel, “Model-based conformance testing of WS-BPEL compositions,” dans *36th Annual IEEE Computer Software and Applications Conference Workshops, COMPSAC 2012, Izmir, Turkey, July 16-20, 2012*, 2012, p. 452–457. [En ligne]. Disponible: <https://doi.org/10.1109/COMPSACW.2012.86>
- [232] A. Jmal Maâlej, M. Krichen et M. Jmaiel, “Conformance testing of WS-BPEL compositions under various load conditions,” dans *36th Annual IEEE Computer Software and Applications Conference, COMPSAC 2012, Izmir, Turkey, July 16-20, 2012*, 2012, p. 371. [En ligne]. Disponible: <https://doi.org/10.1109/COMPSAC.2012.100>
- [233] M. Krichen et S. Tripakis, “State identification problems for finite-state transducers,” dans *Formal Approaches to Testing and Runtime Verification (FATES-RV’06)*, ser. LNCS. Springer, 2006, to appear.
- [234] M.Krichen et S.Tripakis, “State identification problems for timed automata,” dans *The 17th IFIP Intl. Conf. on Testing of Communicating Systems (TestCom’05)*, ser. LNCS, vol. 3502. Springer, 2005.
- [235] G. Myers, *The art of software testing*. Wiley, 1979.
- [236] M. Krichen et S. Tripakis, “Conformance testing for real-time systems,” *Formal Methods in System Design*, vol. 34, n^o. 3, p. 238–304, 2009.
- [237] N. Bertrand, A. Stainer, T. Jéron et M. Krichen, “A game approach to determinize timed automata,” *Formal Methods in System Design*, vol. 46, n^o. 1, p. 42–80, 2015.
- [238] N. Bertrand, A. Stainer, T. Jéron et M. Krichen, “A game approach to determinize timed automata,” dans *International Conference on Foundations of Software Science and Computational Structures*. Springer, Berlin, Heidelberg, 2011, p. 245–259.

- [239] N. Bertrand, T. Jérón, A. Stainer et M. Krichen, “Off-line test selection with test purposes for non-deterministic timed automata,” *Logical Methods in Computer Science*, vol. 8, n^o. 4, p. 1–33, 2012.
- [240] S. Bensalem, M. Krichen, L. Majdoub, R. Robbana et S. Tripakis, “A simplified approach for testing real-time systems based on action refinement,” dans *ISoLA*, ser. Revue des Nouvelles Technologies de l’Information, vol. RNTI-SM-1. Cépaduès-Éditions, 2007, p. 191–202.
- [241] C. N. Ip et D. L. Dill, “State reduction using reversible rules,” dans *Proceedings of the 33rd Conference on Design Automation, Las Vegas, Nevada, USA, Las Vegas Convention Center, June 3-7, 1996.*, 1996, p. 564–567. [En ligne]. Disponible: <https://doi.org/10.1145/240518.240625>
- [242] C. N. Ip, “Generalized reversible rules,” dans *Proceedings of the Second International Conference on Formal Methods in Computer-Aided Design*, ser. FMCAD ’98. London, UK, UK: Springer-Verlag, 1998, p. 403–420. [En ligne]. Disponible: <http://dl.acm.org/citation.cfm?id=646185.758715>
- [243] Chih-Chun Lee, J. R. Jiang, Chung-Yang Huang et A. Mishchenko, “Scalable exploration of functional dependency by interpolation and incremental sat solving,” dans *2007 IEEE/ACM International Conference on Computer-Aided Design*, Nov 2007, p. 227–233.
- [244] J.-H. R. Jiang et R. K. Brayton, “Functional dependency for verification reduction,” dans *Computer Aided Verification*, R. Alur et D. A. Peled, édit. Berlin, Heidelberg: Springer Berlin Heidelberg, 2004, p. 268–280.
- [245] L. Benalycherif et A. McIsaac, “A semantic condition for data independence and applications in hardware verification,” *Electronic Notes in Theoretical Computer Science*, vol. 250, n^o. 1, p. 39 – 54, 2009, proceedings of the Seventh International Workshop on Automated Verification of Critical Systems (AVoCS 2007). [En ligne]. Disponible: <http://www.sciencedirect.com/science/article/pii/S1571066109003296>
- [246] L. Momtahan, “Towards a small model theorem for data independent systems in alloy,” *Electronic Notes in Theoretical Computer Science*, vol. 128, n^o. 6, p. 37 – 52, 2005, proceedings of the Fourth International Workshop on Automated Verification of Critical Systems (AVoCS 2004). [En ligne]. Disponible: <http://www.sciencedirect.com/science/article/pii/S1571066105002355>

BIBLIOGRAPHY

- [247] T. Wahl et A. Donaldson, “Replication and abstraction: Symmetry in automated formal verification,” *Symmetry*, vol. 2, n^o. 2, p. 799–847, 2010.
- [248] M. Kwiatkowska, G. Norman et D. Parker, “Symmetry reduction for probabilistic model checking,” dans *Computer Aided Verification*, T. Ball et R. B. Jones, édit. Berlin, Heidelberg: Springer Berlin Heidelberg, 2006, p. 234–248.
- [249] R. A. Thacker, K. R. Jones, C. J. Myers et H. Zheng, “Automatic abstraction for verification of cyber-physical systems,” dans *Proceedings of the 1st ACM/IEEE International Conference on Cyber-Physical Systems*, ser. ICCPS ’10. New York, NY, USA: ACM, 2010, p. 12–21. [En ligne]. Disponible: <http://doi.acm.org/10.1145/1795194.1795197>
- [250] M. Lahami, M. Krichen et M. Jmaïel, “Safe and Efficient Runtime Testing Framework Applied in Dynamic and Distributed Systems,” *Science of Computer Programming (SCP)*, vol. 122, n^o. C, p. 1–28, 2016.
- [251] M. Lahami, M. Krichen, H. Barhoumi et M. Jmaïel, “Selective test generation approach for testing dynamic behavioral adaptations,” dans *Testing Software and Systems - 27th IFIP WG 6.1 International Conference, ICTSS 2015, Sharjah and Dubai, United Arab Emirates, November 23-25, 2015, Proceedings*, 2015, p. 224–239. [En ligne]. Disponible: https://doi.org/10.1007/978-3-319-25945-1_14
- [252] M. Lahami, M. Krichen et M. Jmaïel, “Runtime Testing Approach of Structural Adaptations for Dynamic and Distributed Systems,” *International Journal of Computer Applications in Technology (IJCAT)*, vol. 51, n^o. 4, p. 259–272, 2015.
- [253] M. Lahami, F. Fakhfakh, M. Krichen et M. Jmaïel, “Towards a TTCN-3 Test System for Runtime Testing of Adaptable and Distributed Systems,” dans *Proceedings of the 24th IFIP WG 6.1 International Conference Testing Software and Systems (ICTSS’12)*, 2012, p. 71–86.
- [254] J. Kienzle, N. Guelfi et S. Mustafiz, *Transactions on Aspect-Oriented Software Development VII: A Common Case Study for Aspect-Oriented Modeling*. Springer Berlin Heidelberg, 2010, ch. Crisis Management Systems: A Case Study for Aspect-Oriented Modeling, p. 1–22.

- [255] R. Mahmud, K. Ramamohanarao et R. Buyya, “Latency-aware application module management for fog computing environments,” *ACM Trans. Internet Technol.*, vol. 19, n^o. 1, p. 9:1–9:21, nov. 2018. [En ligne]. Disponible: <http://doi.acm.org/10.1145/3186592>
- [256] M. Barcelo, A. Correa, J. Llorca, A. M. Tulino, J. L. Vicario et A. Morell, “Iot-cloud service optimization in next generation smart environments,” *IEEE Journal on Selected Areas in Communications*, vol. 34, n^o. 12, p. 4077–4090, Dec 2016.
- [257] A. J. Maâlej, M. Lahami, M. Krichen et M. Jmaïel, “Distributed and resource-aware load testing of WS-BPEL compositions,” dans *ICEIS (2)*. SciTePress, 2018, p. 29–38.
- [258] M. Lahami, M. Krichen, M. Bouchakwa et M. Jmaiel, “Using knapsack problem model to design a resource aware test architecture for adaptable and distributed systems,” dans *ICTSS*, ser. Lecture Notes in Computer Science, vol. 7641. Springer, 2012, p. 103–118.
- [259] C. Guerrero, I. Lera et C. Juiz, “A lightweight decentralized service placement policy for performance optimization in fog computing,” *Journal of Ambient Intelligence and Humanized Computing*, vol. 10, n^o. 6, p. 2435–2452, Jun 2019. [En ligne]. Disponible: <https://doi.org/10.1007/s12652-018-0914-0>
- [260] L. Yang, J. Cao, G. Liang et X. Han, “Cost aware service placement and load dispatching in mobile cloud systems,” *IEEE Transactions on Computers*, vol. 65, n^o. 5, p. 1440–1452, May 2016.
- [261] H. R. Arkian, A. Diyanat et A. Pourkhalili, “Mist: Fog-based data analytics scheme with cost-efficient resource provisioning for iot crowdsensing applications,” *Journal of Network and Computer Applications*, vol. 82, p. 152 – 165, 2017. [En ligne]. Disponible: <http://www.sciencedirect.com/science/article/pii/S1084804517300188>
- [262] B. Ottenwalder, B. Koldehofe, K. Rothermel et U. Ramachandran, “Migcep: Operator migration for mobility driven distributed complex event processing,” dans *Proceedings of the 7th ACM International Conference on Distributed Event-based Systems*, ser. DEBS ’13. New York, NY, USA: ACM, 2013, p. 183–194. [En ligne]. Disponible: <http://doi.acm.org/10.1145/2488222.2488265>

BIBLIOGRAPHY

- [263] O. Skarlat, S. Schulte, M. Borkowski et P. Leitner, “Resource provisioning for iot services in the fog,” dans *2016 IEEE 9th International Conference on Service-Oriented Computing and Applications (SOCA)*, Nov 2016, p. 32–39.
- [264] S. Filiposka, A. Mishev et K. Gilly, “Community-based allocation and migration strategies for fog computing,” dans *2018 IEEE Wireless Communications and Networking Conference (WCNC)*, April 2018, p. 1–6.
- [265] Z. Tang, X. Zhou, F. Zhang, W. Jia et W. Zhao, “Migration modeling and learning algorithms for containers in fog computing,” *IEEE Transactions on Services Computing*, p. 1–1, 2018.
- [266] Z. Wen, R. Yang, P. Garraghan, T. Lin, J. Xu et M. Rovatsos, “Fog orchestration for internet of things services,” *IEEE Internet Computing*, vol. 21, n^o. 2, p. 16–24, Mar 2017.
- [267] H. Zhang, Y. Xiao, S. Bu, D. Niyato, F. R. Yu et Z. Han, “Computing resource allocation in three-tier iot fog networks: A joint optimization approach combining stackelberg game and matching,” *IEEE Internet of Things Journal*, vol. 4, n^o. 5, p. 1204–1215, Oct 2017.
- [268] V. B. Souza, X. Masip-Bruin, E. Marin-Tordera, W. Ramirez et S. Sanchez, “Towards distributed service allocation in fog-to-cloud (f2c) scenarios,” dans *2016 IEEE Global Communications Conference (GLOBECOM)*, Dec 2016, p. 1–6.
- [269] H. Gupta, A. V. Dastjerdi, S. K. Ghosh et R. Buyya, “ifogsim: A toolkit for modeling and simulation of resource management techniques in the internet of things, edge and fog computing environments,” *Softw., Pract. Exper.*, vol. 47, n^o. 9, p. 1275–1296, 2017. [En ligne]. Disponible: <https://doi.org/10.1002/spe.2509>
- [270] J. Wu, S. Luo, S. Wang et H. Wang, “Nles: A novel lifetime extension scheme for safety-critical cyber-physical systems using sdn and nfv,” *IEEE Internet of Things Journal*, vol. 6, n^o. 2, p. 2463–2475, 2018.
- [271] Z. Guan, Y. Zhang, L. Wu, J. Wu, J. Li, Y. Ma et J. Hu, “Appa: An anonymous and privacy preserving data aggregation scheme for fog-enhanced iot,” *Journal of Network and Computer Applications*, vol. 125, p. 82–92, 2019.

BIBLIOGRAPHY

- [272] D. G. Greenspan, “Ending the bitcoin vs blockchain debate <http://www.multichain.com/blog/2015/07/bitcoin-vs-blockchain-debate/>,” *MultiChain*, 2015.
- [273] A. Shomer, “The colored coins protocol,” <https://github.com/Colored-Coins/Colored-Coins-Protocol-Specification/wiki>, 2015, [Online; Available].
- [274] A. Maria, Z. Aviv et V. Laurent, “Hijacking bitcoin: Routing attacks on cryptocurrencies,” <https://arxiv.org/pdf/1605.07524v2.pdf>, 2017.
- [275] S. Abeyratne et R. Monfared, “Blockchain ready manufacturing supply chain using distributed ledger,” *International Journal of Research in Engineering and Technology*, 2016.
- [276] J. Umeh, “Blockchain double bubble or double trouble?” *Itnow*, vol. 58, n^o. 1, p. 58–61, 2016.
- [277] M. Löf, “Decentralized transactions in a centralized environment: A blockchain study within the transport industry,” 2017.
- [278] “Open Web Application Security Project (OWASP),” https://www.owasp.org/index.php/Main_Page, 2021, [Online; Available].
- [279] Z. Guan, Y. Zhang, L. Wu, J. Wu, J. Li, Y. Ma et J. Hu, “Appa: An anonymous and privacy preserving data aggregation scheme for fog-enhanced iot,” *Journal of Network and Computer Applications*, vol. 125, p. 82 – 92, 2019. [En ligne]. Disponible: <http://www.sciencedirect.com/science/article/pii/S1084804518303060>
- [280] A. Lahbib, K. Toumi, A. Laouiti, A. Laube et S. Martin, “Blockchain based trust management mechanism for iot,” dans *2019 IEEE Wireless Communications and Networking Conference (WCNC)*. IEEE, 2019, p. 1–8.
- [281] R. Jabbar, M. Kharbeche, K. Al-Khalifa, M. Krichen et K. Barkaoui, “Blockchain for the internet of vehicles: a decentralized iot solution for vehicles communication using ethereum,” *Sensors*, vol. 20, n^o. 14, p. 3928, 2020.
- [282] “MythX: Smart contract security service for Ethereum . [Online]. Available: <https://mythx.io/>,” 2020.

BIBLIOGRAPHY

- [283] “Smart Contract Security Verification Standard .” <https://github.com/securing/SCSVS>, 2020, [Online; Available].
- [284] M. Ge, H. Bangui et B. Buhnova, “Big data for internet of things: a survey,” *Future generation computer systems*, vol. 87, p. 601–614, 2018.
- [285] O. Nasraoui et C.-E. B. N’Cir, “Clustering methods for big data analytics,” dans *Techniques, Toolboxes and Applications*. Springer, 2019, p. 192.
- [286] E. Karafiloski et A. Mishev, “Blockchain solutions for big data challenges: A literature review,” dans *IEEE EUROCON 2017-17th International Conference on Smart Technologies*. IEEE, 2017, p. 763–768.
- [287] Z. Wang, Y. Tian et J. Zhu, “Data sharing and tracing scheme based on blockchain,” dans *2018 8th international conference on logistics, Informatics and Service Sciences (LISS)*. IEEE, 2018, p. 1–6.
- [288] L. Yue, H. Junqin, Q. Shengzhi et W. Ruijin, “Big data model of security sharing based on blockchain,” dans *2017 3rd International Conference on Big Data Computing and Communications (BIGCOM)*. IEEE, 2017, p. 117–121.
- [289] D. Fang, Y. Qian et R. Q. Hu, “Security for 5g mobile wireless networks,” *IEEE Access*, vol. 6, p. 4850–4874, 2017.
- [290] D. Fang et Y. Qian, “5g wireless security and privacy: Architecture and flexible mechanisms,” *IEEE Vehicular Technology Magazine*, vol. 15, n^o. 2, p. 58–64, 2020.
- [291] A. M. Alwakeel, A. K. Alnaim et E. B. Fernandez, “A survey of network function virtualization security,” dans *SoutheastCon 2018*. IEEE, 2018, p. 1–8.
- [292] G. S. Aujla, M. Singh, A. Bose, N. Kumar, G. Han et R. Buyya, “Blocksdn: Blockchain-as-a-service for software defined networking in smart city applications,” *IEEE Network*, vol. 34, n^o. 2, p. 83–91, 2020.
- [293] S. Zhang, “An overview of network slicing for 5g,” *IEEE Wireless Communications*, vol. 26, n^o. 3, p. 111–117, 2019.

- [294] B. Nour, A. Ksentini, N. Herbaut, P. A. Frangoudis et H. Moun gla, “A blockchain-based network slice broker for 5g services,” *IEEE Networking Letters*, vol. 1, n^o. 3, p. 99–102, 2019.
- [295] V. A. Siris, D. Dimopoulos, N. Fotiou, S. Voulgaris et G. C. Polyzos, “Trusted d2d-based iot resource access using smart contracts,” dans *2019 IEEE 20th International Symposium on “A World of Wireless, Mobile and Multimedia Networks”(WoWMoM)*. IEEE, 2019, p. 1–9.
- [296] H. Cui, Z. Chen, N. Liu et B. Xia, “Blockchain-driven contents sharing strategy for wireless cache-enabled d2d networks,” dans *2019 IEEE International Conference on Communications Workshops (ICC Workshops)*. IEEE, 2019, p. 1–5.
- [297] L. Xie, Y. Ding, H. Yang et X. Wang, “Blockchain-based secure and trustworthy internet of things in sdn-enabled 5g-vanets,” *IEEE Access*, vol. 7, p. 56 656–56 666, 2019.
- [298] V. Adat, I. Politis, C. Tselios, P. Galiotos et S. Kotsopoulos, “On blockchain enhanced secure network coding for 5g deployments,” dans *2018 IEEE Global Communications Conference (GLOBECOM)*. IEEE, 2018, p. 1–7.
- [299] V. Ortega, F. Bouchmal et J. F. Monserrat, “Trusted 5g vehicular networks: Blockchains and content-centric networking,” *IEEE Vehicular Technology Magazine*, vol. 13, n^o. 2, p. 121–127, 2018.
- [300] S. Sharma, R. Miller et A. Francini, “A cloud-native approach to 5g network slicing,” *IEEE Communications Magazine*, vol. 55, n^o. 8, p. 120–127, 2017.

Résumé : Le concept de villes intelligentes gagne de plus en plus en importance dans les métropoles modernes en raison de l'émergence et de la diffusion d'appareils, de systèmes et de technologies intelligents embarqués et connectés dans la vie quotidienne, qui ont créé l'opportunité de connecter chaque «chose» à Internet. Dans l'ère à venir de l'Internet des objets, l'Internet des véhicules (IoV) jouera un rôle crucial dans la construction d'une ville intelligente. En fait, l'IoV a le potentiel de résoudre efficacement divers problèmes de trafic. Il est essentiel pour améliorer l'utilisation des routes, réduire la consommation d'énergie et la pollution et améliorer la sécurité routière. Néanmoins, le principal problème concernant l'IoV, et en particulier le Véhicule-à-Véhicule (V2V) et le Véhicule-à-infrastructure (V2I), est l'établissement de paiements et de communications sécurisés et instantanés. Pour répondre à ce défi, ce travail propose une solution basée sur la Blockchain pour mettre en place un paiement et une communication sécurisés afin d'étudier l'utilisation de la Blockchain comme middleware entre différents acteurs des systèmes de transport intelligents. Dans cette étude, nous avons évalué les propriétés les plus importantes de la solution développée, à savoir la consommation de la mémoire et de l'énergie, l'immutabilité, la confidentialité, la cohérence, l'intégrité, le temps d'exécution et le coût. L'objet de cette évaluation est de s'assurer de la capacité de la plateforme basée sur la Blockchain à assurer une communication efficace et un paiement sécurisé avec l'IoV. Selon les résultats, cette plateforme peut contribuer à résoudre les défis les plus critiques de la communication véhicule-à-tout (V2X) en améliorant la sécurité et l'évolutivité .

Mots clés : Blockchain, Internet des Véhicules, Communication Automobile, Parking Intelligent, Paiements Automatisés, Ethereum, Ville Intelligente, Système de Transport Intelligent, Cloud et Android.

Abstract : The concept of smart cities is increasingly gaining prominence in modern metropolises due to the emergence and spread of embedded and connected smart devices, systems, and technologies in everyday lives, which have created an opportunity to connect every 'thing' to the Internet. In the upcoming era of the Internet of Things, the Internet of Vehicles (IOV) will play a crucial role in constructing a smart city. In fact, the IOV has a potential to solve various traffic problems effectively. It is critical for enhancing road utilization, reducing energy consumption and pollution, and improving road safety. Nevertheless, the primary issue regarding the IoV, and in particular to Vehicle-to-Vehicle (V2V) and Vehicle-to-Infrastructure (V2I), is establishing secure and instant payments and communications. To respond to this challenge, this work proposes a Blockchain-based solution for establishing secure payment and communication in order to study the use of Blockchain as middle-ware between different participants of intelligent transportation systems. The proposed framework employs Ethereum to develop a solution aimed at facilitating Vehicle-to-Everything (V2X) communications and payments. Moreover, this work qualitatively test the performance and resilience of the proposed systems against common security attacks. Computational tests showed that the proposed solution solved the main challenges of Vehicle-to-X (V2X) communications such as security and centralization.

Keywords : Blockchain, Internet of Vehicles, Automotive communication, Smart Parking, Automated Payments, Ethereum, Smart City, Intelligent Transport System, Cloud and Android.

