Blockchain for Incentive Insurance and ITS Related Payment Services for Automated and Connected Vehicles

A final report submitted to

Central Institute of Technology, Kokrajhar

in partial fulfilment for the award of the degree of

Bachelor of Technology

in

Computer Science & Engineering

by

Roshan Singh, Gwmsrang Muchahary, Mridutpal Lahon

(Gau-c-15/L-297, Gau-C-15/069, Gau-c-15/L-300)

Under the supervisions of

Pranav Kumar Singh

Dept. of Computer Science & Engineering

Central Institute of Technology, Kokrajhar

Even Semester, 2019

April 26, 2019

DEPT. OF COMPUTER SCIENCE & ENGINEERING

CENTRAL INSTITUTE OF TECHNOLOGY, KOKRAJHAR

KOKRAJHAR - 783370, INDIA

CERTIFICATE

This is to certify that the project report entitled "**Blockchain for Incentive Insurance and ITS Related Payment Services for Automated and Connected Vehicles**" submitted by **Roshan Singh, Gwmsrang Muchahary, Mridutpal Lahon** (Roll No. Gau-c-15/L-297, Gau-C-15/069, Gau-c-15/L-300) to Central Institute of Technology, Kokrajhar towards partial fulfilment of requirements for the award of degree of Bachelor of Technology in Computer Science & Engineering is a record of bona fide work carried out by him under my supervision and guidance during Even Semester, 2019.

Date: April 26, 2019
Place:

Pranav Kumar Singh
Dept. of Computer Science & Engineering
Central Institute of Technology,
Kokrajhar
Kokrajhar - 783370, India

Acknowledgements

We have taken efforts in this project. However, it would have been possible without the kind support and help of the Computer Science and Engineering Department (Central Institute of Technology, Kokrajhar). I would like to extend my sincere thanks to all of them.

We are highly indebted to Pranav Kumar Singh, Department of Computer Science and Engineering, for his guidance and constant supervision as well as for providing necessary

Contents

Certificate i

Acknowledgements ii

Contents iii

List of Figures v

List of Tables vi

Abbreviations vii

1 Introduction 1
- 1.1 Vehicular Plane . . . 2
  - 1.1.1 BSM . . . 2
  - 1.1.2 PSM . . . 2
- 1.2 RSU Plane . . . 3
- 1.3 Communications . . . 3
  - 1.3.1 V2V (Vehicle to Vehicle Communications) . . . 3
  - 1.3.2 V2I (Vehicle to Infrastructure) . . . 4
  - 1.3.3 V2X (Vehicle to Anything) . . . 4
- 1.4 Blockchain . . . 5
  - 1.4.1 Public Blockchain: . . . 6
  - 1.4.2 Permissionless blockchain: . . . 7
  - 1.4.3 Private Blockchain: . . . 7
  - 1.4.4 Permissioned Blockchain: . . . 7
- 1.5 Smart Contracts . . . 8
- 1.6 Ethereum Framework . . . 8
  - 1.6.1 Externally Owned Accounts: . . . 9
  - 1.6.2 Contract Accounts: . . . 9

2 Motivation and Objectives 10

3 Related Works 12

4 Blockchain Based ITS Related Services: 14
4.1 System Architecture: . . . 14
4.2 Framework . . . 15
4.3 Implementation . . . 15
4.3.1 Decentralized Intelligent Parking System (DIPS) . . . 15
4.3.2 Decentralized Tolltax Payment System (DTTPS) . . . 19

5 Usage Based Insurance:(UBI) 20
5.1 UBI . . . 20
5.2 Challenges in traditional Insurance Processes . . . 21
5.3 How Blockchain can solve these challenges . . . 21
5.4 Telematics . . . 22
5.5 Proposed Blockchain based framework for UBI . . . 22
5.6 Involved Entities in our System . . . 25
5.7 Transaction . . . 26
5.8 Offline Transaction Signing: . . . 28

6 Incentives 31
6.1 The Need for a Token . . . 31
6.2 ERC20 Standard . . . 32
6.3 Introducing W4-Coin . . . 33

7 Experiments 34
7.1 DIPS and DTTPS . . . 34
7.2 Telematics . . . 35
7.2.1 RSU Configuration: . . . 36
7.2.2 Vehicle OBU Configuration: . . . 36
7.3 Programming Constructs . . . 36

8 Results and Discussion 38
8.0.1 Comparing Average Execution Time: . . . 38
8.0.2 Our approach v/s Centralized approach . . . 39

9 Conclusion 41

Bibliography 42

List of Figures

4.1 Proposed System Architecture 15
4.2 System framework of our implemented system 16

5.1 Proposed Framework 24
5.2 System Diagram for Offline Transaction Signing 29

6.1 ERC20 Standard 32

7.1 Vehicle OBU 35
7.2 RSU 35
7.3 Proposed Framework 35
7.4 A sample access modifier 37

List of Tables

Abbreviations

BC	**B**lockchain
CV	**C**onnected **V**ehicle
FBI	**F**ederal **B**ureau of **I**nvestigation
UBI	**U**sage **B**ased **I**nsurance

Chapter 1

Introduction

Vehicular Ad Hoc Network (VANET)(1) is a classification of MANET (Mobile Adhoc Network). Unlike MANET the nodes in a VANET are highly mobile. A typical VANET consists of vehicles, Road Side Units (RSUs) and the Traffic Authority(TA). The vehicles are the mobile nodes which keeps their position changing over time. RSUs are considered to be the infrastructure deployed by the Traffic Authority. RSUs are responsible for providing the nodes a range of services such as ITS related and infotainment services. TA is the authorized entity which is considered to be the supreme authority who oversees the management as well as the maintenance of the infrastructure. TA is also responsible for setting up the initial configurations of the nodes to enable them to participate in the VANET. As the node in the VANET keeps on changing its position over time it comes in the range of different RSUs. These nodes are equipped with sophisticated hardware device known as On Board Units (OBUs) which let them communicate with other mobile nodes as well as the infrastructure on the network. Based on the location where the operations are performed the working of the VANET can be categorized into two planes:

1.1 Vehicular Plane

The vehicular plane consists of mobile nodes or the vehicles. The network topology at the vehicular plane keeps on changing. At the vehicular plane the vehicles communicate with each other in order to share certain information. The vehicular plane also includes entities other than the vehicles such as a pedestrian or a cyclist. The sharing of the information is achieved by the means of exchange of certain messages. Depending upon the criticality of the exchanged messages the messages can be categorized as

1.1.1 BSM

BSM stands for Basic Safety message. These are the core V2V messages exchanged by the vehicles with their peers to inform them about an urgent prevailing situation around its neighbourhood. An example of a BSM is Forward Collision Warning (FCW) message. FCW is an active safety warning feature that warn the drivers for an imminent frontal collision. The alert is raised when a vehicle comes too close in front of another vehicle.

1.1.2 PSM

PSM stands for Personal Safety Message. These are the messages broadcasted by the Vulnerable Road User (VRU) devices to indicate their presence on the road.PSM helps to avoid accidents where the VRU is beyond the visibility of the driver of an approaching vehicle.

1.2 RSU Plane

The RSU plane consists of the RSUs and the infrastructure deployed by the TA. At the RSU plane each RSU is connected with each other with the help of dedicated lines. The communication among these RSUs are considered to be secure and fast. The main goal of the deployment of these RSUs by the TA is to provide instant access to information to the vehicles operating at the vehicular plane. The propagation of information at this plane is considered to be fast. Also, the TA can directly communicate with the entities on this plane.

1.3 Communications

Depending upon the entities participating in the communication process in the VANETs the entire communication domain can be modelled as:

1.3.1 V2V (Vehicle to Vehicle Communications)

The V2V communication takes place at the Vehicular plane. The entities participating in V2V are the only the vehicles. The vehicles communicate with each other with the help of a standard protool named DSRC.

DSRC or the Dedicated Short Range Communication is a short range to medium range wireless communication channel specially designed for vehicular communications.It works at 5.9 GHz band. The DSRC can operate within the range of 10 - 1000 meters. It provides a high bandwidth rate suitable for implementation for fast data dissemination at the Vehicular plane.

1.3.2 V2I (Vehicle to Infrastructure)

The V2I communications takes place between the vehicles at the vehicular plane and among the RSUs/Infrastructures working at the RSU plane. Like V2V here also the communication medium is wireless, however the protocol to be implemented varies upon the requirements of the deployer. Now a days communication technologies such as LTE and 5G are being tested for as choice of protocol for V2I communications. An example for V2I communication can be a vehicle requesting for an infotainment service to the RSU.

1.3.3 V2X (Vehicle to Anything)

The V2X communication takes place between the vehicles on the road and any other non transportation entity such as a pedestrian. V2X communication can include communication between vehicle and driver smart phone etc.

Due to its long term prospects VANET is turning out to be a hot topic among the research community as well as in the industrial domain. The core objective for implementing VANETs is to make the vehicles communicate with each other in order to reduce the number of road accidents, improve the traffic efficiency and to provide drivers a better driving experience. The current approaches of providing the ITS related services are neither efficient nor reliable.

With this in mind, a blockchain based approach for Decentralized Intelligent Parking System and Decentralized Intelligent Toll tax payment is developed in this work. The rest of the work is organized as follows. We first define our problem statement. We then discuss the related work carried out in the domain till date. We continue with giving a brief introduction to blockchain technology the underlying technology

used in our work and describing our proposed mechanism, the evaluation of the results of the experiments performed on our developed model. And finally conclude with discussing future prospects with our developed model.

1.4 Blockchain

Blockchain can be considered as one of the cutting edge technology of 21st century. Blockchain is a new technology which is based upon some core principles in Computer Science and Mathematics that are in existence from past hundreds of years such as cryptography. A blockchain as its name suggest is essentially a sequence of blocks where each block is linked with its previous block with the help of hashes. A block is formed of zero or more data elements known as transactions. The transactions are signed by the entity executing it. PKI is used for signing in and verifying the transactions on the blockchain. A blockchain is maintained by a set of nodes where each node maintain its own local copy of the chain. These nodes communicate with each other with the help of P2P connections. The blockchain provides some key features which makes it a bit unique from other technologies in Computer Science.One of them is decentralization a blockchain can never be owned or can be dominated by a single entity. Theoretically, it is possible but practically it is not. To own a blockchain one has to get control over at least 51 percent of the nodes maintaining the chain. Second key feature is immutability data once written on the chain is immutable and cannot be erased. Blockchain can be used to establish trust among a set of untrusted nodes. The blockchain assumes the nodes in the network to be untrusted. It utilises a mechanism to make all these untrusted nodes to agree on a same state. This mechanism is known as consensus. The consensus mechanism may vary depending upon the type of blockchain to be considered. In case of public

blockchain where anyone can come and be a part of the network the common consensus mechanism used is Proof of Work (PoW). There exists certain special nodes on the network which collects the transactions coming from various clients on the network and bundle them together to form a block. These miners need to perform a computationally complex operation to get their block on to the chain. This computationally complex problem is termed as PoW. It is difficult to generate but easy to check. Since the miners invest their computational resources for mining the blocks they also get incentivised by the network in terms of mining rewards. Depending upon the type of the blockchain the consensus algorithm to be used varies. It is suggested to use challenge response based consensus algorithms such as PoW for public blockchains where the entities are unknown and unauthenticated. Consensus algorithms such as PBFT and PoA can be used in case of private blockchains. Each consensus algorithm have their own advantages and drawbacks. Such as the PoW algorithm can scale well in terms of the number of nodes, however it provides a very low throughput in terms of transaction processing. In the bitcoin blockchain (2) the average throughput is 7 tx/s. On the other hand algorithms like PBFT can scale pretty well in terms of transaction throughput however it fails miserably when coming to scalability in terms of number of nodes.

Based upon the read and write permissions on the chain a blockchain can be categorised into four categories:

1.4.1 Public Blockchain:

A public blockchain is a blockchain which is open to everyone. Anyone can read the data on the chain. Anyone can join the network and can download full copy of the entire network and can run node in their local devices. Anyone in the world can

send transaction to anyone and can see transactions details in public explorer. Examples: Bitcoin, Litecoin, Monero etc.

1.4.2 Permissionless blockchain:

A permissionless blockchain is a blockchain which does not have any restriction on writing data onto it. Anyone is allow to create their own address and can begin interact with the network by submitting a transaction on it.

1.4.3 Private Blockchain:

A private blockchain only permits a set of users to view the data present on the chain. A private blockchain is run by specific the member of the consortiums and companies. Examples of private blockchains are Hyperledger Fabric, Multichain, Monax.

1.4.4 Permissioned Blockchain:

A permissioned blockchain is a blockchain which puts a set of restrictions in order to decide which set of entities will be able to write data onto the chain.

The first application of the blockchain technology was the cryptocurrency Bitcoin . Bitcoin is a peer to peer cryptocurrency which lets user transact on a peer to peer network in an unrestricted manner. Bitcoin is a public, permissionless blockchain which allows anyone to join and transact on it. Blockchain platforms such as Bitcoin are meant for dedicated purpose that means they can be used only for those tasks for which they are made. Like the Bitcoin blockchain only supports the financial transactions which involves transferring of bitcoins from one account to another.

And such platforms also doesn't provide much programming flexibility in terms of writing and executing turing complete scripts.

However, there exists blockchain platforms such as Ethereum and Hyperledger which permits users to write and deploy their own turing complete codes.

1.5 Smart Contracts

The smart contracts(3) are a bunch of self-executable code sitting on top of a blockchain. A smart contract consists of a well defined set of rules/condition and a set of actions. The conditions specify the actions to be performed when it turns out to be true. The property of self execution makes a smart contract much more unique than any other programming construct. A smart contract works on the data present on the blockchain. A smart contract is also immutable that means once deployed on the chain the rules of the contract cannot be changed. This property of smart contract ensures bias free decisions and helps to minimize the need of trust. It can be used as a tool for automating decision making process without the need of an intermediary in a blockchain environment.

1.6 Ethereum Framework

Ethereum(4) is a public blockchain platform. Unlike Bitcoin blockchain Ethereum supports smart contracts. That means snippets of code can be written and deployed on Ethereum blockchain. The supportability of smart contracts makes the platform ideal for developers who want to implement their smart contract for certain problem

specific use case. The Ethereum blockchain is based upon a State Transition based model, where each node executes a set of transactions in the same order to reach a common consensus. Ether is a cryptocurrency, which is used as a medium of exchange on the Ethereum blockchain. The platform uses the notion of GAS(a unit of measuring the computational complexity) and GAS LIMIT while processing the transactions.The idea of GAS is used to curb the menace of an attacker who can embed an infinite loop in his smart contract and executes a transaction. In the absence of GAS and GAS LIMIT it will lead to a halting problem in the blockchain and may result in a crash of the chain. The Ethereum blockchain supports two types of accounts:

1.6.1 Externally Owned Accounts:

These are the accounts owned by the users on the blockchain. A externally owned account is associated with a set of public/private key pairs. These are used by the user to sign and verify the transactions.

1.6.2 Contract Accounts:

These are the accounts where the code of the smart contract resides. An externally owned account can send and receive ether to and from other externally owned accounts or contract accounts. The contract accounts cannot do anything of its own. It depends upon the transactions executed from the externally owned accounts which triggers the execution of the code in present in the contract account.

Chapter 2

Motivation and Objectives

The objective of the work performed in this project is motivated by the need to have a decentralized and stable ITS related services scheme for VANETs. The earlier approaches are mainly node centric where a single authority was responsible for providing the services and managing the happening in order to provide that service in VANET. Such schemes make the entire system dependent on a single entity which is not monitored by any other entity. This can lead to an ample scope of discrepancies that could happen in the system if that centralized authority becomes un-available or does not provide the services.

Our work tries to address this challenge by proposing a blockchain based decentralized approach for implementing ITS related services in VANETs. In our work we implement a use case of an ITS related service to be used by the vehicles.

In this work we also propose a blockchain based framework for UBI. Unlike the traditional vehicular insurance paradigm where the user policy premium is decided by the insurance company based upon the past history. Our framework aims to harness the potentials of UBI where the policy premium will be decided based upon

the current or recent behaviours of the driver. With the introduction of blockchain in our framework we aim to make the proceedings more transparent and smooth.

Chapter 3

Related Works

There have been quite a few works in the domain of blockchain based vehicular insurance. Authors in (5) describe how Blockchain can be used for making liability decision. They mentioned and claimed the use of operational and decision partitioned communication in their proposed framework ensuring secure data exchange based only on a need to know basis. A telematics-based approach for vehicular insurance is proposed in (6). The work describes how telematics can introduce a new model of billing for insurance premiums. The authors argue that the use of telematics data can help determine the policy holder's insurance premium. A study of risks and implementation challenges of vehicular insurance has been done in (7) (8). In (9) author study motor fraud risks using empirical evidence. A study on possibilities and limitations of smart contract based vehicular communication is provided in (10). The authors implement a blockchain based ITS service for gas refueling. In (11) authors propose and implement a parking management scheme with real-time parking navigation and anti-theft protection. A cloud-based smart vehicle parking system is proposed in (12). Authors propose a cryptocurrency based secure incentive scheme for VANETs in (13).

In(14) authors propose a Machine Learning based, Histogram Oriented Gradient in Intelligent Transportation System approach for alerting drivers about the frontal presence of obstacles as well as pedestrian. A J2EE based architecture GIS application is proposed in (15) for fast access of web pages in VANETs.

However, approaches using decentralized technologies such as blockchain is very much deficient in the domain of implementing ITS related services and vehicular insurance. We consider that much more contributions are needed in this domain using such decentralized technologies. To, this end our approach will be one such effort.

Chapter 4

Blockchain Based ITS Related Services:

This chapter provides a detailed explanation about the framework, working logic and implementation details of our blockchain based decentralized ITS services namely Decentralized Parking System (DIPS) and Decentralized Toll Tax Payment System (DTTPS).

4.1 System Architecture:

The proposed system architecture to facilitate our blockchain based strategy for implementing incentive and ITS related services is shown in Figure 7.3. We have considered two major plane of the architecture those are the Vehicular plane and the RSU plane. The Vehicular plane consists of smart vehicles. The RSU plane includes Traffic Authority (TA), RSUs and ITS services.

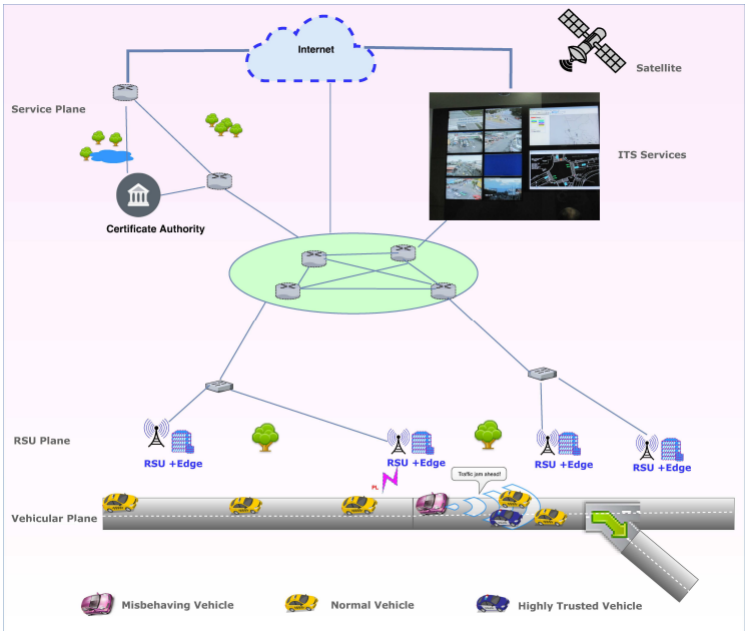


FIGURE 4.1: Proposed System Architecture

4.2 Framework

The system framework of our implemented system is as shown as in Figure 4.2. The TA deploys the contract on the blockchain. RSUs interact with the contract via Command Line Interface. RSU uses Node.js API for communicating with the blockchain. On the other hand vehicles are provided with Graphical User Interface. They communicate with the blockchain with the help of Web3.js API.

4.3 Implementation

4.3.1 Decentralized Intelligent Parking System (DIPS)

To fulfill the objective of implementing a DIPS, our smart contract stores the details of available parking zones in an area. A vehicle can search for available parking zone in a particular area by using the PIN/ZIP code associated with that particular area.

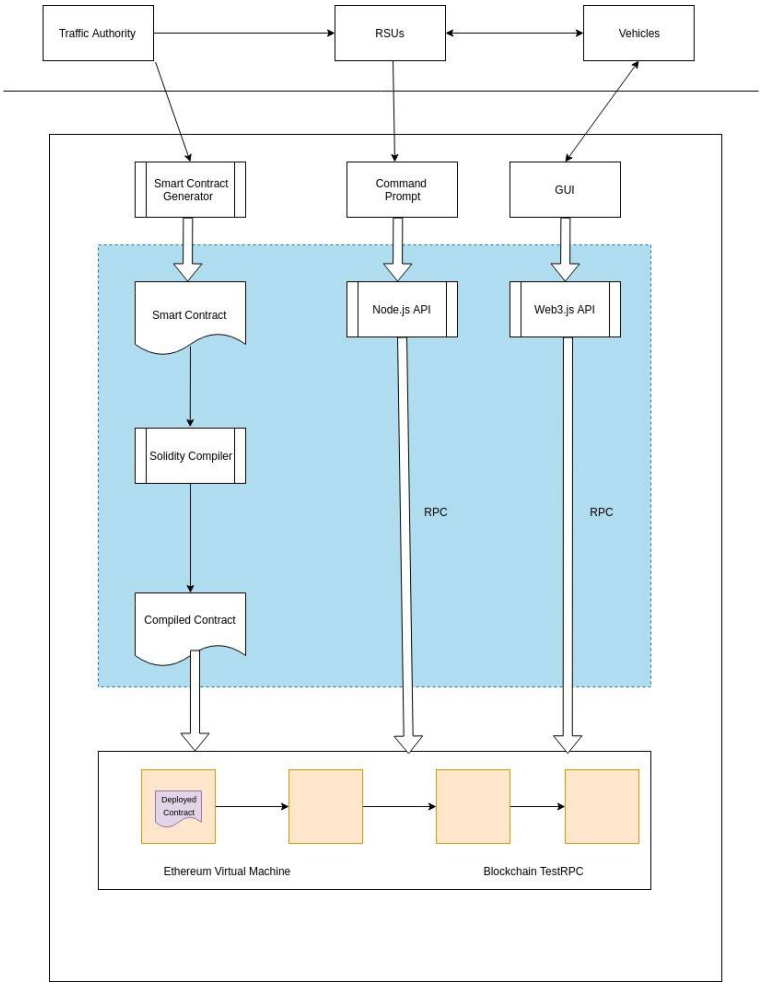


FIGURE 4.2: System framework of our implemented system

We have used the PIN code as an unique identifier here. The Traffic Authority (TA) adds a Parking Zone into the blockchain by calling the method *configureParkingZone* as shown in figure . A vehicle can poll the blockchain for getting the list of available parking zones available in a particular area. Our contract provides a getter function called *checkAvailiability* for this.

The method also displays the rate of parking charged by each of these parking zones. A vehicle can book a lot in the parking zone by calling the function *bookMyLot* as shown in Figure . Figure shows the process flow involved in booking a lot.

On being called method results in a successful transaction if there exists enough

parking lots available in the parking zone, otherwise the transaction gets reverted. In case of a successful transaction the vehicle is allocated with a lot identifier determining the parking lot allocated to it in a given parking zone. The smart contract maintains a counter to specify the duration of time the parking lot was occupied by the vehicle. Each parking lot is equipped with sensing devices which senses the ID of the vehicle trying to park in the lot. If the address of the parking vehicle matches with the address of the vehicle which has booked the lot then only the vehicle is allowed to park itself in the lot otherwise not. The counter starts when the vehicle books a lot and gets over when the vehicle exists the lot. The vehicle can exit a lot by calling the function *exitParking* as shown in Figure

Upon resulting in a successful transaction the counter associated with the vehicle gets off and the parking lot occupied by the vehicle in the contract gets free. The vehicle can now pay the parking bill. The vehicle first checks the bill incured for parking by calling the function *parkingBill* as shown in Figure . It displays the amount of ether to be paid as a bill. The parking bill is calculated with the formula:

$$ParkingBill = Durationof Parking * Rateof Parking \tag{4.1}$$

After getting the parking bill the vehicle can pay the bill by calling the method *payMyBill* as the amount to be paid displayed by the function. It is to be noted that if a vehicle exists a parking lot and does not pays the bill the vehicle will be unable to book further more slots in some later point of time.

Our developed smart contract for DIPS also provides the facility of cancelling a lot booked on the blockchain. It provides a method called *cancelBooking* as shown in Figure for this. However, to deal with the notorious vehicles trying to book and cancel the lots uselessly we have taken certain preventive measures.

- A booked slot can only be cancelled by a vehicle who has booked it. We specify a threshold time that specify the duration during which the cancellation of the booked lot can be made. There exists no scope of cancelling a booked lot at some later point of time once the threshold duration has passed.

- To discourage any vehicle for frequent cancelling of lots we also imposed a cancellation charge.

Our DIPS smart contract also facilitate the Parking Lot Manager to adjust the parking charges. However, to curb the menace of any greedy Parking Lot Manager who frequently increase as well as decrease the rates, we have taken certain preventive measures here.

- To discourage any vehicle for frequent cancelling of lots we also imposed a cancellation charge.

A Parking Lot Manager has to notify the changes in the rates from a particular date before implementing the changes. If the Parking Lot Manager has not notified earlier he will not be able to implement the changes.

We developed a Web based application for the vehicles to interact with the DIPS. Figure c shows the web based interface at the vehicle end for booking a parking lot. The drivers can book a slot by using the Web based interface. A web based real time portal was also developed for the Parking Lot Manager to view the status of its parking lot.

4.3.2 Decentralized Tolltax Payment System (DTTPS)

We also implemented a decentralized toll tax payment system, a ITS related service . Private blockchain handle data flow based on the smart contract or authorized policies set up by the TA. Like our previous implementation here also we use the cryptocurrency ether which is provided in the core of the ethereum client as a medium of exchange.Incentives earned by the smart vehicles can be redeemed as ether which can be utilized by them for accessing various ITS related services. The procedure for our implemented ITS service is shown in Figure . The vehicle identity is fetched from the vehicle with some sensors present at the toll crossing. The device installed at toll crossing checks the block-chain for assuring that the vehicle has paid the tax, if yes the indicator installed at the toll crossing go green indicating the vehicle to cross.

Chapter 5

Usage Based Insurance:(UBI)

5.1 UBI

UBI sometimes known as Pay-As-You-Drive is a type of vehicle insurance where the cost of the premium paid by the driver is dependent upon the usage of the vehicle. It is different from the traditional vehicle insurance in the sense that unlike the traditional methods where the liability of the driver is determined based on the driving duration or the past driving history here the liability of a driver is determined by factors such as his driving behavior including braking actions, acceleration rate, and another usage factor. The data are collected from the vehicles with the help of telematics devices installed on board. UBI for vehicles helps insurance company identifying safe and careless drivers and reward the safe one by giving them lower premiums encouraging them to continue their driving habits.

Vehicular insurance has a billion dollars market and is growing every year. However, it is facing a lot of challenges such as fraudulent claims and cheating from insurance companies. The traditional centralized insurance framework is not at all suitable for

the vehicular environment. We consider the UBI(Usage Base Insurance) model as a baseline of our work. At the same time, we are working on incentive related issues.

5.2 Challenges in traditional Insurance Processes

After going through different papers(7) (9) (8) and articles, we have find out some of the challenges that vehicular insurance companies are facing these days.

- Involvement of fraudulent activity is so huge that even the police and legal authorities fails to identify exact correct information.
- Because of its centralized architecture system, there is a high risk of cyber-related attacks.
- Lot of paperwork is need to be done in every step, so it consumes lot of time.
- Involvement of different third party agents.

5.3 How Blockchain can solve these challenges

According to a report by Federal Bureau of Investigation (FBI) 2015, the total costs of insurance fraud (non-health insurance) is estimated to be more than $40 billion per year.But traditional insurance fails to solve and identify these frauds. Followings are the some of reasons that could reduce above fraud activity & challenges.

- Because of its immutable characteristics, no one could make changes data and claim for it.
- It is visible to everyone, ie transparency.

- Smart contract powered by blockchain technology could provide customers to manage claims in a transparent, responsive and automatic manner.
- Fraud detection and risk prevention. Vehicular and claim detailed are stored in blockchain, Fraud/Criminal find it more difficult to hide their identity or attempt to claim more than once. The blockchain technology reduces paperwork and provide easier access to data. The transparency and verifiable help to reduce fraud.
- Blockchain is a decentralized system. So instead of depending on central server, it uses peer to peer architecture. In case central server goes down it will still work.

5.4 Telematics

Telematics, in a broad sense, refers to any integrated use of telecommunication with information and communication technology including transporting, wireless communication and global positioning system. It is the technology of sending and receiving and storing information relating to remote object via telecommunication device.

5.5 Proposed Blockchain based framework for UBI

The figure shows our proposed blockchain based framework for UBI. The processes in the system goes as - A vehicle before coming onto the road registers itself with the Traffic Authority. The traffic authority is a trusted authority which is also responsible for providing a set of certificates to the intelligent vehicles. An intelligent vehicle uses these certificates for their V2V and V2I communications. Once on

the road an intelligent vehicle sends updates on its system status in the form of blockchain transactions. These updates can be either of these two: a) Periodic Updates : Such as a vehicle sending its carburetor data or the fuel level of the vehicle. b) Event triggered Updates : Such as harsh braking, rush driving, sharp turn etc.

The insurance company can periodically monitor the driving behaviours of a vehicle and can decide to lease the premium amount or to impose a penalty on the insured driver. The system basically helps the insurance company to analyze the risks of an insured customer prior any misfortune happens.

Whenever an IV meets an accident or is involved in an accident the vehicle is seized and the legal authoritative proceedings takes place. The legal authority verifies the accident spot and prepares a formal report of the accident. The legal authority logs an accident event onto the blockchain with the details as such as the involved vehicles, location, time and other relevant information. The legal authority as well as the Insurance company requests the TA to disclose the identity of the vehicle. Once, the identity of the vehicle is disclosed the legal authority can poll the blockchain to find out the causes of the accident by analyzing the last few transactions of the involved vehicles. Also, the Insurance company analyses a set of behavioural aspects of the vehicle and then decides whether to give the incentives to that vehicle or not.

In case, the vehicle need a repairing after an accident. The vehicle can be brought to one of the service center registered onto the blockchain. Once, the vehicle has been repaired the insurance company/ the customer can pay the bills via the blockchain. As each of the transactions will be recorded onto the blockchain it will reduce the risks of fraud from the service center as it may charge higher prices from the customers.

The manufacturer can utilize our blockchain based framework for disseminating several firmware updates. A manufacturer can put an update of the availability of a new firmware update along-with the hash of the update file. An IV which often polls the blockchain will find the availability of the update and will install the update. Since, the hash of the update will be available onto the blockchain a vehicle can always check the integrity and authenticity of the downloaded update. Thus, our system also prevents dissemination of malicious contents embedded into the firmware updates.

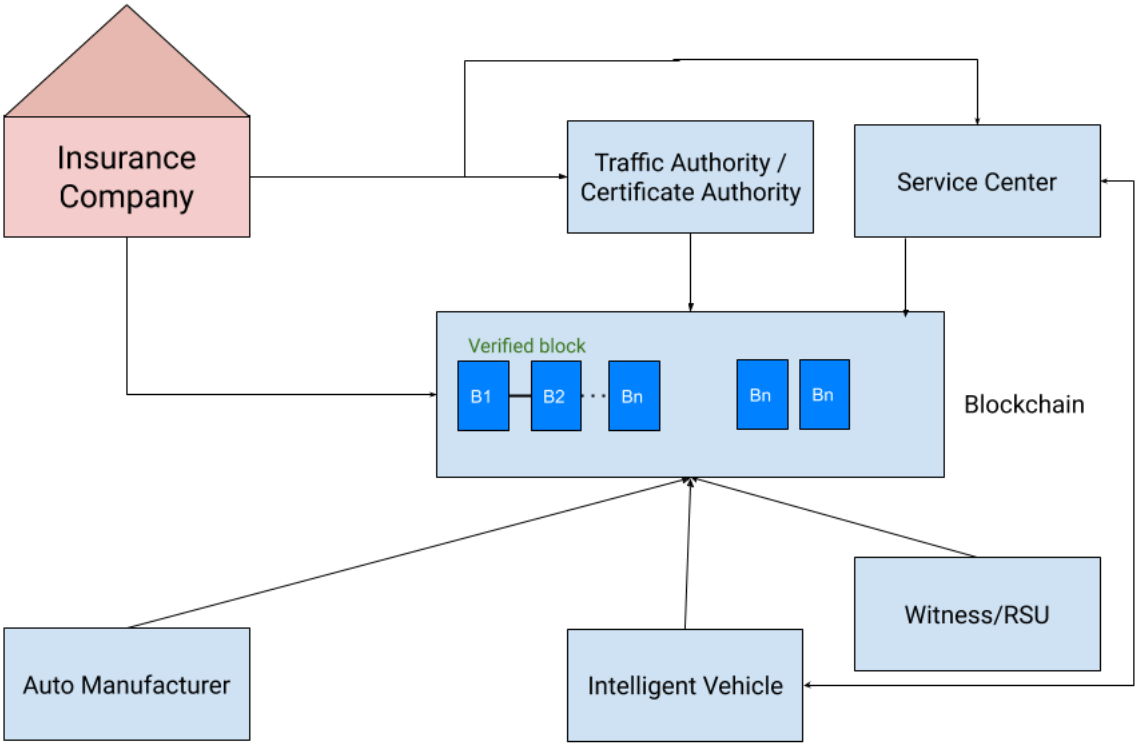


FIGURE 5.1: Proposed Framework

5.6 Involved Entities in our System

1. Intelligent Vehicle (IV) : It provides necessary details of evidence to make liability decisions. A black box unit installed in or OBU, On Board Unit is assumed to be temper resistance storage unit where data is generated by sensors such as speed, location, GPS, time of accident, video and pictures data of accident. It also able to make hash of that data for data integrity. A telematics device installed on vehicles diagnostic port is able to send these details to blockchain and smart contract is responsible for collection to process it.

2. Auto Manufacturer (AM) : Responsible for storing vehicles sensor details to keep tracks of maintenance report. It also provides necessary information to vehicles like software update, bug report, notification etc.

3. Witness Involved (WI): Nearby Roadside Units provides necessary details like video and picture only on request from Insurance Company via the Traffic Authority.

4. Insurance Company (IC) : Receives evidence from Intelligent Vehicles, Auto Manufacturer, Legal Authorities and from Traffic Authority, process it to make liability decision. Provides compensation to IV if claim is meeting terms and conditions as well provides reward based on UBI report.

5. Legal Authority (LA) : Settles the disputes if occurs. Basically it compromises of court and police. Receive evidence data from manufacturer, Insurance Company and Witness Involves. Investigate it and provides proof of a suitable evidence to Insurance Company.

6. Traffic Authority (TA) : Provides necessary evidence like picture and video of event to Insurance Company as well as to Legal Authority. A RSU can be Traffic Authority (TA).

   We refer police and court as Legal Authorities (LAs) as they are believe to be most prominent way to proof an evidence if third party is involved. Figure described the interactions between entities in the proposed liability model.

5.7 Transaction

In Blockchain transactions are the communication between entities. Any data that is sent to the Blockchain is assumed to be transaction and in our proposed framework vehicles are sending sensor details as transactions. In our framework, transaction could be single as well as multiple sign transaction called mutiSig transaction. Signing only single signature is called single sign transaction and multiple is called multiSig transaction. A multiSig transaction ensures that multiple signature as valid transaction. Followings are the transactions we considered :

- **StartTripTransaction (STT) :** Single sign transaction that is generated by Intelligent vehicles when each vehicle started to drive. On each start of trip, vehicles will initiate this transaction and sent it to the Blockchain. The structure is given as follows.

$$\mathbf{STT} = [(TID)T_s(S_{data})(Sig)]$$

TID : Is the transaction id which is different for every transaction.

S_{data} **:** It contains data like speed, location, position, accelerator, harsh braking

T_s : It is the time of event occurrence.

Sig : Single signature of vehicle.

- **Event Triggered Transaction (ETT) :** When sensors in vehicles detected some conditions like multiple harsh braking is applied, slippery roads, over speeding on in traffic jams, bumpy roads, this transaction is generated by Autonomous vehicles that will be single sign transaction. Structure is as below.

$$\mathbf{ETT} = [(TID)T_s(S_{data}(ETT_{data})(P_{data})](Sig)$$

ESM_{data} : Event Safety Message (ESM) contains data like number of harsh brake applied, speed, location etc.

P_{data} : This is the last data in temper proof of storage before event. Contains data like hash of video and picture capture during accident occurred. It also store the actual video and picture data in locally.

- **Primary Evidence Transaction (PET) :** Generated by Intelligent vehicles when accident occurs. Structure is give as below.

$$\mathbf{PET} = [(TID)T(_s)E_(data)ESM_(data)P_(data)(Sig)]$$

E_{data} : Generated by Intelligent vehicles when accidents occurs. Structure is given as follows.

$$E_{(data)} = [(Loc)T(_s)ESM_(data)H(E_{data})]$$

- **SecondaryEvidenceTransaction (SET) :** Each Intelligent vehicles generated two evidence transactions, first is PET and second is SET. This will used if PET is fail to proof accident. In case in collision between two vehicles it can be act as witness for involving vehicles. Structure is as follows.

$$\textbf{SET} = [(TID)T(_s)ESM_(data)(Sig)]$$

ESM_{data} : Contains hash of video and picture and related footage that is captured by front/back cameras. Original contents is also store in vehicles locally storage device.

- **Periodic Update Transaction (PUT) :** Generated by Intelligent vehicles every after 1 month. Structure is as follows.

$$\textbf{PUT} = [(TID)T(_s)S(_data)(Sig)]$$

S_{data}**:** It contains data like tire pressure, fuel level, kilometers driven, average speed. Since we focus more on UBI, collection of driver behavior and finding out his driving character is one of the first priority.

5.8 Offline Transaction Signing:

Every transaction executed on the blockchain is needed to be signed by the private key of the sender. Often the private key is stored on the device running the client software. In case of an intelligent vehicle the OBU of the vehicle is a device responsible for the interaction with the blockchain. Also, the OBU itself contains the private key. However, the OBU is not only communicating with the blockchain but at the same time it is communicating with the various RSUs in its path for accessing a lot of services such as infotainment services and traffic related services. Such an open interface leaves an ample scope for the OBU to be compromised by the hackers and malicious entities which may lead to the compromise of the private key further.

To tackle this problem we have introduced the idea of offline-transaction signing by the intelligent vehicles. The intelligent vehicle can store the private keys

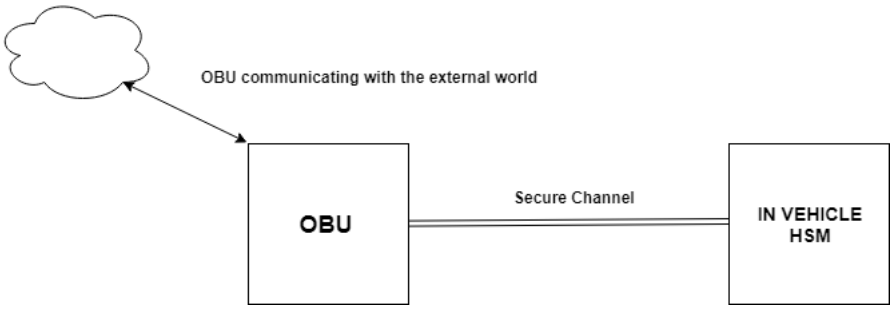


FIGURE 5.2: System Diagram for Offline Transaction Signing

on a separate device known as the (Hardware Safety Module)HSM device. The HSM device is not open to communication to the external world. The HSM communicates with the OBU via a fast secure channel. Whenever, the OBU needs to execute a transaction it can send the transaction to be signed by the HSM. The HSM after signing the transaction can sent the transaction to the OBU which can broadcast the transaction to the blockchain network to get it mined.

Algorithm 1 Reward and Premium algorithm

N : Intelligent Vehicle (IV)
S : Set of all IV belongs to Insurance Company
p : A parameter that belongs to F
threshold : Maximum limit of acceptance $\forall P \in F$
cutoff: Minimum score required to receive reward and premium discount

Require: $IV_i.score \leftarrow 0$
for each IV_i **in S**
for each p in S
if $IV_i.P \leq threshold$.p **then**
$IV_i.score \leftarrow IV_i.score + 1$
else
$IV_i.score \leftarrow IV_i.score - 1$
end if
endfor
if $IV_i.score \leq cutoff$ **then**
rewardToken($IV_i.address, IV_i.score$)
lowerPremium($IV_i.address$)
else
highPremium($IV_i.address$)
end if
endfor

Chapter 6

Incentives

Incentives given to an intelligent vehicle in VANETs plays an important role for motivating the vehicle to behave well by making them obey the driving rules thus maintaining the harmony in the traffic. The incentives earned by a vehicle can be directly utilized by the vehicle for accessing and consuming various intelligent transportation services one such can be the usage of our developed decentralized parking system. In our system the incentives are provided by the TA and the Insurance company. The TA provides the incentives in the form of tokens whereas the insurance company provides incentives in the form of providing premium discounts to the vehicles behaving continuously well.

6.1 The Need for a Token

As our entire framework is based upon the Ethereum blockchain which is a general blockchain platform that supports implementation of multiple use cases such as financial transactions, IoT related transactions, logistics transactions etc. At the end of the day our system is going to be deployed onto the Ethereum main-net

blockchain which deals with real valuable ether. Depending directly on the ether as the medium of exchange will make our system dependent on the performance of other systems running on the blockchain. This will not only hamper the economy of our system at the same time it can result in less number of participants utilizing our system.

6.2 ERC20 Standard

ERC20 is a protocol standard that defines certain rules for issuing token on Ethereum's network. A token can be buy, sell and trade. Ethereum tokens are simply digital assets that are being built on top of the Ethereum blockchain. In 'ERC20', ERC stands for Ethereum Request For Comments and 20 stands for a unique ID number to distinguish this standard from others. Similar to like we have an HTTP protocol for internet, we have a standard protocol for tokens to be issued on Ethereum i.e. ERC20. Any token that is issued on ERC20 standard must uses these six function in their smart contract program as given in figure 6.1 .

```
1 // ----------------------------------------------------------------------------
2 // ERC Token Standard #20 Interface
3 // https://github.com/ethereum/EIPs/blob/master/EIPS/eip-20-token-standard.md
4 // ----------------------------------------------------------------------------
5 contract ERC20Interface {
6     function totalSupply() public constant returns (uint);
7     function balanceOf(address tokenOwner) public constant returns (uint balance);
8     function allowance(address tokenOwner, address spender) public constant returns (uint remaining);
9     function transfer(address to, uint tokens) public returns (bool success);
10    function approve(address spender, uint tokens) public returns (bool success);
11    function transferFrom(address from, address to, uint tokens) public returns (bool success);
12
13    event Transfer(address indexed from, address indexed to, uint tokens);
14    event Approval(address indexed tokenOwner, address indexed spender, uint tokens);
15 }
```

FIGURE 6.1: ERC20 Standard

6.3 Introducing W4-Coin

In our work we created our own token named W4-Coin which will act as a medium of incentive for consuming and providing the services in our system. In the previous sections we mentioned our DIPS and DTTPS services where we accepted ether as the medium of exchange. The same services can be accessed with the W4-Coin too. Instead, we can also offer certain discounts on the services which are accessed using the RCoin over the ether.

Chapter 7

Experiments

The following sections describes the details of each of the various test-bed experiments performed in the project.

7.1 DIPS and DTTPS

To implement our decentralized approach for DIPS and DITPS with the help of smart contract, we use a private permission Ethereum blockchain. The backbone connectivity between the RSUs are wired network. The connectivity between Vehicles and RSUs are wireless. For writing and compiling the contract, we used the Remix integrated development environment(IDE) and for Solidity a browser-based IDE. We performed experiments where the Raspberry Pi 3 node (vehicle) sends a set of the transaction of type bookMyLot and payTollTax to our private blockchain platform in an asynchronous manner, i.e., all transactions are sent without waiting for a response from the blockchain. The number of requests has been set to 1, 100, 500, 750 and 1000. The results of the experiment averaged over 3 independent runs.

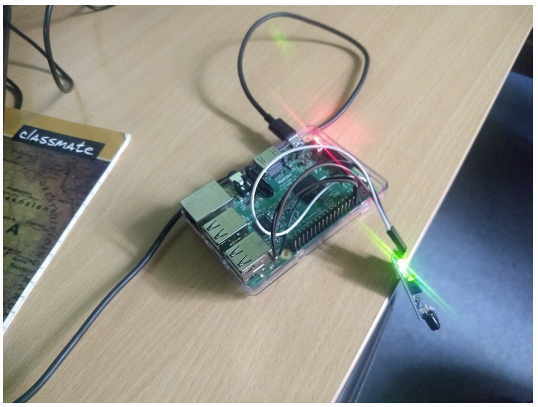


FIGURE 7.1: Vehicle OBU

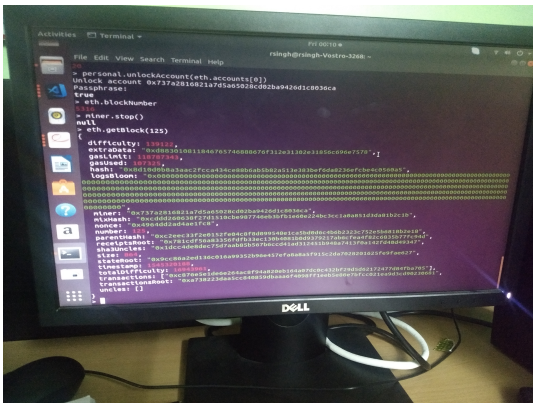


FIGURE 7.2: RSU

7.2 Telematics

We implemented detection of harsh braking as a part of implementing the usage of telematics in our blockchain based decentralized UBI.

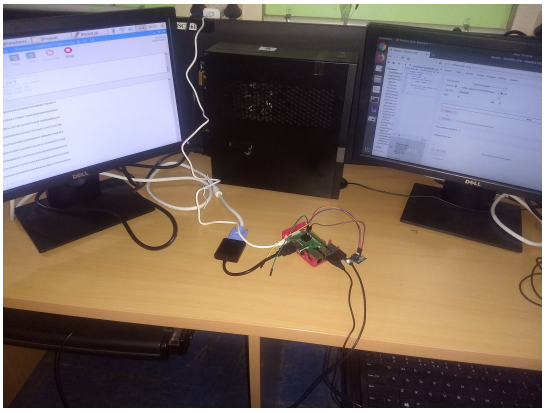


FIGURE 7.3: Proposed Framework

We used Raspberry Pi 3 B+ as the vehicle OBU. We mimicked the braking behaviour of a vehicle with the help of an infrared sensor. Where when the IR sensor detects an obstacle it assumes that the brakes are applied. The RPi executes a transaction whenever it finds that the brakes have been applied. The transaction signing mechanism was offline as described earlier. We wrote a smart contract that monitors the speed of the vehicle. We write the logic that was able to detect a sudden change or

a sudden deceleration of the vehicle. When such a change is detected a transaction is logged stating that hard brakes has been applied.

The testbed details are as follows:

7.2.1 RSU Configuration:

- Intel CoreTM i7-7700 CPU 3.60GHz x 8, 8 GB RAM
- 1 TB hard drive
- Ubuntu 18.04.1 LTS Operating System
- Geth Version : geth 1.8.17-stable

7.2.2 Vehicle OBU Configuration:

- Raspberry Pi 3.3/ Raspberry Pi 3 B+
- Raspbian Operating System
- Geth Version : geth 1.8.18 ARMv7 stable release

7.3 Programming Constructs

One of the characteristics of a smart contract is its immutable nature. It's impossible to make any kind of modification in a deployed contract. This fact is often overlooked by many smart contract developers which in turn results in heavy financial losses. In our case also management of incentives can be considered as management of valuable asset which are valuable for the vehicles as far as their credit score is concerned.

In order to make our smart contract bug free as well as secure we have developed the contract as per as the guidelines provided in the official documentation of the language used. We have also considered core principles of best practices in the development of smart contracts.

The code of the contract has been tested a number of times. A major portion of the project time line was devoted for this purpose. Both static as well as run time analysis were done. The detected bugs were fixed and appropriate access restrictions were kept in place where necessary. Some of the modifiers used in our smart contract are as follows :

```
modifier onlyTrafficAuthority{
    bool auth = false;
    if(msg.sender == trafficAuthority_Address){
        auth = true;
    }
        require(auth == true,"You are not the Traffic Authority");
        _;
}
```

FIGURE 7.4: A sample access modifier

For instance the modifier shown in Figure 7.4 specifies that only a Traffic Authority can execute a function where the modifier onlyTrafficAuthority is imposed.

Chapter 8

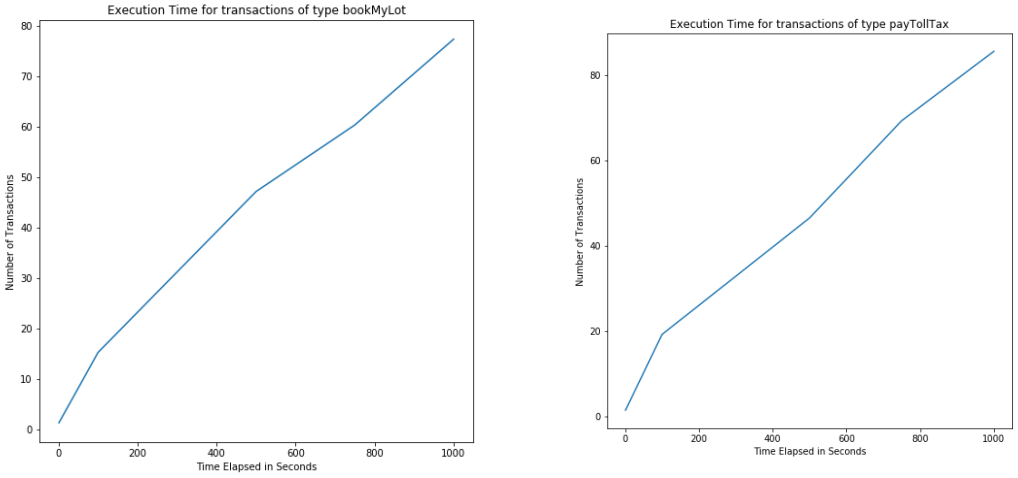
Results and Discussion

We choose transaction execution time as the parameter for the evaluating our set up private blockchain.

- Execution Time: The execution time is the total amount of time (number of seconds) during which the blockchain took to execute and confirm all transactions in the dataset. It is the duration of time elapsed, when the first transaction was deployed to the time when the last transaction is mined. Average execution time for a set of transaction is the execution time averaged over certain independent run.

8.0.1 Comparing Average Execution Time:

We compare the differences in execution time of varying transactions, we executed the transactions in the batch of 1 100 500 750 and 1000 transactions. The execution time grows as the number of transaction in the data set increases. For a batch of 1000 transactions the blockchain took 78.33 seconds to execute transactions of type *bookMyLot* and 79.6 for the transactions of type *payTollTax*. The execution time



plot for*bookMyLot* is as shown in Figure 5.0.1 and the execution time plot for *payTollTax* is as shown in Figure. **??**

As we can see from the average execution time plot increases exponentially for larger set of transactions. This is a drawback of the used blockchain platform. The Ethereum blockchain fails to grow linearly for a large set of transactions with its default Ethash PoW based Algorithm.

8.0.2 Our approach v/s Centralized approach

- Transparency and Immutability: All the proceedings either it may be booking a parking lot or paying the toll tax are commited on the blockchain as in the form of mined transactions. With, our developed scheme, it is impossible for the vehicle to cross the toll without making a payment.Similarly, it is also not possible for the Parking Lot Manager to deny the fact that a vehicle has not paid for parking or to charge higher rates of parking. All the code in the smart contract are visible publicly. So anyone can check it.

- Availability: As compared to centralized approach our approach always guarantee of a availability as long as a full node exist in the system. Traditional centralized approaches does not guarantee this characteristic. This ensures that one can always do a transaction using our scheme.

- Trust : In traditional approaches the parties involved in a transaction needed to trust each other. These parties are often highly untrusted and tries to cheat each other. Like the Parking Lot Manager always aims to charge a higher rate of parking to increase his profit. We minimize this trust relationship to a maximum extent. Saying, truly we have just removed the need of trust here.

Chapter 9

Conclusion

In this work we proposed and implemented two blockchain based ITS related services namely Decentralized Toll Tax Payment System (DTTPS) and Decentralized Intelligent Parking System (DIPS). We implemented both the services on the Ethereum blockchain platform with a test-bed implementation. We made an performance analysis of the system with a experimental study. As a telematics application for our usage based vehicular insurance we implemented detection of harsh braking with a prototype setup. We recorded the harsh braking event on the blockchain as a transaction. We also proposed a blockchain based framework for usage based vehicular insurance. We incorporated the logic of providing incentives to the vehicles in our system into the framework. We created our own token that can be used as a medium of exchange for accessing various services in our system.

This is one of the initial projects in the domain which aims to solve some critical challenges with a novel and yet evolving technology. We plan to extend the work further. The whole system integration along with the implementation of claim processing we keep these as our future work.

Bibliography

[1] H. Hartenstein, K. Laberteaux, VANET: vehicular applications and internetworking technologies, Vol. 1, Wiley Online Library, 2010.

[2] S. Nakamoto, et al., Bitcoin: A peer-to-peer electronic cash system.

[3] N. Szabo, Formalizing and securing relationships on public networks, First Monday 2 (9).

[4] G. Wood, et al., Ethereum: A secure decentralised generalised transaction ledger, Ethereum project yellow paper 151 (2014) 1–32.

[5] R. J. S. J. Chuka Oham, Salil S.Kanhere, A blockchain based liability attribution framework.

[6] I. F. M. M. Sinisa Husjjak, Dragon Perakovic, Telematics system in usage based motor insurance, Peer-review under responsibility of DAAAM Internation Vienna 100 (2014) 816–825.

[7] D. I. Tselentis, G. Yannis, E. I. Vlahogianni, Innovative motor insurance schemes: A review of current practices and emerging challenges, Accident Analysis & Prevention 98 (2017) 139–148.

[8] J.-A. Tarr, Distributed ledger technology, blockchain and insurance: opportunities, risk, and challenges, Insurance Law Journal 29 (2018) 254–268.

[9] D. N. Kiragu, Drivers of motor vehicle insurance fraud risk: Empirical evidence from insurance companies in kenya.

[10] Y. Hanada, L. Hsiao, P. Levis, Smart contracts for machine-to-machine communication: Possibilities and limitations, in: 2018 IEEE International Conference on Internet of Things and Intelligence System (IOTAIS), IEEE, 2018, pp. 130–136.

[11] R. Lu, X. Lin, H. Zhu, X. Shen, Spark: A new vanet-based smart parking scheme for large parking lots, in: IEEE INFOCOM 2009, IEEE, 2009, pp. 1413–1421.

[12] Q. G. K. Safi, S. Luo, L. Pan, W. Liu, R. Hussain, S. H. Bouk, Svps: Cloud-based smart vehicle parking system over ubiquitous vanets, Computer Networks 138 (2018) 18–30.

[13] Y. Park, C. Sur, K.-H. Rhee, A secure incentive scheme for vehicular delay tolerant networks using cryptocurrency, Security and Communication Networks 2018.

[14] LiFei, LiDengLin, Pedestrian detection based on histogram of oriented gradient in intelligent transport system.

[15] X. Lu, Web based public participation gis service for intelligent transportation information collection.