BLOCKCHAIN BASED PSEUDONYM MANAGEMENT SCHEME FOR INTELLIGENT TRANSPORT SYSTEM

This thesis is submitted in partial fulfillment of the requirement for the award of the degree in Ph.D. in Computer Science.

University of Peshawar

SUBMITTED BY

SUMAIRA JOHAR

Ph.D. SCHOLAR

SUPERVISED BY

DR. NAVEED AHMAD

DEPARTMENT OF COMPUTER SCIENCE

UNIVERSITY OF PESHAWAR

(November 2013)

BLOCKCHAIN-BASED PSEUDONYM MANAGEMENT SCHEME FOR INTELLIGENT TRANSPORT SYSTEM

Approval Certificate

This is to certify that the research work presented in this thesis, entitled **"BLOCKCHAIN BASED PSEUDONYM MANAGEMENT SCHEME FOR INTELLIGENT TRANSPORT SYSTEM"** is conducted by Ms. Sumaira Johar under the supervision of Dr. Naveed Ahmad.
No part of this thesis has been submitted anywhere else for any other degree. This thesis is submitted to the Department of Computer Science, University of Peshawar in partial fulfillment of the requirements for the degree of Ph.D. in Computer Science.

Student Name: Sumaira Johar Signature: _____
Ph.D. Scholar

Supervisor:

Dr. Naveed Ahmad Signature: _____
Designation: Assistant. Professor
Department of Computer Science
University of Peshawar, Pakistan.

Examination Committee

a) Thesis Evaluator and Examiner-I
Name: Dr. Nadeem Iqbal Signature: _____
Designation: Associate Professor
Department of Computer Science
Abdul Wali Khan University Peshawar

b) Examiner-II
Name: Dr.Bilal Jan Signature: _____
Designation: Assistant Professor
Department of Computer Science
FATA University Kohat

Internal Examiner:

Name: Dr. Azhar Rauf Signature: _____
Designation: Professor
Department of Computer Science
University of Peshawar, Pakistan

Chairman:

Prof. Dr. Shah Khusro Signature: _____
Department of Computer Science
University of Peshawar, Pakistan.

Research Declaration

I hereby announce that my thesis entitled "Blockchain-based Pseudonym Management Scheme for Intelligent Transport System" is authentic research. Moreover, all the information, facts, and results are thoroughly cited and fully referenced.

Sumaira Johar

Rights of the Thesis

DEDICATED TO

My Respected Parents

For their spiritual support, sacrifices, guidance, and prayers. I wish my parents were alive to see my hard work and have appreciated seeing my goal achieved which was their dream too. Your hard work and wise guidance have provided me with successful academic and social careers.

My Siblings

For their motivation, and encouragement. You have provided a cooperative and pleasant environment where I can freely express my financial, academic, and social problems.

My Husband and Kids

For their support and encouragement. Without my husband's cooperation, this would not have been possible. My husband and kids are a source of motivation and inspiration for me in my life.

I am grateful to all the members of my family, who are inspiration and motivation for me. This thesis would not be possible without their support.

Acknowledgment

Firstly, I am highly fortunate to have Almighty Allah's blessings that enable me to complete my Ph.D. thesis.

My gratitude and amiable thanks go to my thesis supervisor Dr. Naveed Ahmad for his support, guidance, and motivation. He is my mentor and his guidance enlightened my path towards achieving this degree.

I would like to thank the honorable chairman Dr. Shah Khusro, Dr. Saeed Mahfooz, Dr. Muhammad Abid, and Dr. Azhar Rauf for their expert comments, suggestions, and encouragement that enable me to refine my Ph.D. research. Moreover, I thank all the staff of the Department of Computer Science, for their clerical and technical support throughout my Ph.D. research.

I thank the National Center for Cyber Security (NCCS) for awarding the stipend for my Ph.D. studies. My Ph.D. would not be possible without the financial support of NCCS.

Also, I thank my fellow lab mates in the "National Center for Cyber Security" initiative for the project "Provable Security of Blockchain Technologies".

In last, I would like to thank my husband, brothers, sister, and my friends for their prayers and spiritual support throughout my Ph.D.

Sumaira Johar

List of Publications

1. Research and Applied Perspective to Blockchain Technology: A Comprehensive Survey

   Accepted and Indexed

   Journal Title: MDPI, Applied Sciences

   Impact Factor: 2.679, HEC category W

   Sumaira Johar, Naveed Ahmad, Warda Asher, Haitham Cruickshank, and Amad Durrani.

   Appl. Sci. 2021, Issue 11, Volume 14, Page(s) 6252

   Digital Object Identifier: https://doi.org/10.3390/app11146252

2. Proof of Pseudonym: Blockchain-Based Privacy Preserving Protocol for Intelligent Transport System

   Accepted and Indexed

   Journal Title: IEEE Access

   Impact Factor: 3.367, HEC category W

   Sumaira Johar, Naveed Ahmad, Amad Durrani and Gauhar Ali

   IEEE Access 2021, Volume 9, Page(s) 163625 – 163639

   Digital Object Identifier: 10.1109/ACCESS.2021.3133423

Table of Contents

Table of Figures

List of Tables

List of Acronym

BC	Blockchain
BFT	Byzantine Fault Tolerance
CAM	Cooperative Awareness Messages
CA	Certificate Authority
CCTV	Closed Circuit Television
DPoS	Delegated Proof of Stack
DBFT	Delegated Byzantine Fault Tolerance
DAG	Directed Acyclic Graph
DSRC	Dedicated Short Range Communication
DDoS	Distributed Denial of Service
DoS	Denial of Service
EMR	Electronic Medical Record
EHR	Electronic Health Record
EVM	Ethereum Virtual Machine
FBA	Federated Byzantine Agreement
IoT	Internet of Things
ITS	Intelligent Transportation System
LPoS	Leased Proof of Stack
OBU	On Board Unit
PoA	Proof of Activity
PoS	Proof of Stake
PoSV	Proof of Stake Velocity

PoST	Proof of Space Time
Proof of Burn	PoB
PoO	Proof of Ownership
PoW	Proof-of-Work
PoX	Proof of Exercise
PoL	Proof of Luck
PoR	Proof of Retrievability
PoE	Proof of Existence
PoC	Proof of Capacity
PoET	Proof of Elapsed Time
PoP	Proof of Publication
PM	Privacy Manager
PKI	Public Key Infrastructure
RSU	Road Side Unit
SCP	Stellar Consensus Protocol
UHF	Ultra-High Frequency
VHF	Very High Frequency
WAVE	Wide Area Virtual Environment
VANETs	Vehicular Adhoc Networks

Abstract

Intelligent Transportation Systems (ITS) is the future for safe and secure transportation. Vehicles in the ITS share basic safety information which can lead to the disclosure of the real identity of the vehicles. Hence adversaries can misuse these safety messages. Pseudonyms are alias granted to vehicles by trusted authorities to conceal their original identities. To avoid linkability and tracking, various pseudonym generation and distribution protocols have been proposed. However, such protocols pose overheads in the system. Therefore, re-using the existing pseudonyms through shuffling is the optimal mechanism for ITS. The Blockchain (BC) is a digital ledger and tamper-resistant record of transactions. To validate the transactions, miners have to implement the blockchain's consensus. The existing shuffling mechanism uses traditional consensus algorithms to support the cryptography operation which leads to overheads in terms of execution time, and memory usage.

This thesis proposed Proof of Pseudonym consensus protocol for the shuffle scheme to improve the efficiency of consensus in terms of execution time and memory. Proof of pseudonym has the idea of electing nodes which is common in Proof of Kernel work but it is not solving the puzzle as is done in Proof of Work or Proof of Kernel Work. Proof of Pseudonym adopted the best features from Proof of Kernel Work and Proof of Elapsed Time. The results show that the proposed Proof of Pseudonym achieves consensus in less time as compared to Proof of Work, Proof of Kernel Work, and Proof of Elapsed Time. The execution time of Proof of Pseudonym is less as compared to other consensus algorithms. Our proposed protocol achieves a better average time as compared to other algorithms. The security and privacy analysis revealed that our proposed protocol achieves identity privacy, unlinkability, and non-repudiation

properties. Threat analysis evaluates the proposed protocol in terms of both internal and external attacks.

Chapter 1 Introduction

This chapter contains a brief introduction about Intelligent transport systems (ITS), Blockchain, privacy issues, and pseudonyms in ITS.

1.1. Intelligent Transport System

Transportation faces many issues over time i.e. traffic congestion, high accidents rate, traffic & carbon emissions causing air pollution, etc. Due to such complexity, researchers came up with the idea of an Intelligent Transport System (ITS) which integrates virtual technology with transportation. ITS is an emerging technology that has many applications like traffic management and congestion control for example [1] vehicles can alert each other in case an accident has occurred somewhere on a road to avoid traffic jams. There are nine integrated components of an ITS. These components include freeway management, traffic signal control, transit management, traveler information services, electronic fare payment, incident management, electronic toll payment, railroad grade crossing safety and emergency management services.

There are numerous methods of wireless communications technologies for intelligent transportation systems. For short and long-range communication Radio communication is used both on Very High Frequency (VHF) and Ultra-High Frequency (UHF).

Protocols of IEEE 802.11 are used for short-range communications of about 350 m, particularly Dedicated Short Range Communications (DSRC) or Wide Area Virtual Environment (WAVE) standard being supported by the Intelligent Transportation Society of America and the United States Department of Transportation. Theoretically, the range of these protocols can be extended using mesh

networking or mobile ad hoc networks. Longer range communications such as Global System for Mobile Communications (IEEE 802.16); WiMAX (IEEE 802.16), or 3G. Unlike short-range communications, these methods require expensive infrastructure for deployment.

1.1.1. Applications of Intelligent Transport System

ITS applications can be divided into the following three categories.

1. **Basic Management Systems:** Such as control systems for a traffic signal; car navigation; container management systems; automatic number plate recognition or speed cameras and variable message signs.
2. **Monitoring Applications**: Such as incident detection, CCTV systems, and detection systems for stopped systems.
3. **Advanced Applications:** That integrates feedback and live data from several sources, such as weather information; parking information and guidance systems; bridge de-icing systems. Moreover, predictive procedures are being established to allow innovative comparison and modeling with historical data.

1.2. Blockchain Technology

The Blockchain (BC) is a digital ledger of transactions that is tamper-resistant. It can be used to record not only monetary transactions [2] but also everything of cost. BC removes the role of trusted third-party as it provides the services of Certificate Authority (CA). Fundamentally, every transaction executed in a system is copied in BC and this copy is available to all those nodes which are in connection with BC. The information in a Blockchain is stored in cryptographically encrypted chunks known as blocks [3]. The next successive block contains information about the previous block

and hence forms a chain. Therefore it is termed a Blockchain. Each block in a block-chain contains a unique hash, transaction data and, a hash of the previous block. The initial block is known as the genesis block. A genesis block does not contain a previous hash. The participants of the blockchain network can be organizations, individuals, or institutions that share a copy of the ledger that contains their valid transactions in a sequential manner [4]. The new transactions are added to the existing records by the consensus of the miners participating in that network. To validate the transactions, miners have to implement the blockchain's algorithm to get rewarded a native token as per existing economic consensus mechanisms like Proof of Work, Proof of Stake, etc.

The miner nodes validate each transaction in a block and add it to the blockchain. Bitcoin is a distributed digital currency, without a central administrator or authority, which can be sent on the peer-to-peer bitcoin network between users without the need for intermediaries. In bitcoin, miner nodes take nearly 10 minutes to validate and add to the blockchain. A miner is selected from a pool of miners using the Proof-of-Work (PoW) consensus mechanism. A blockchain uses a consensus mechanism to allow the miners to agree on a single value. After successful validation by all the miners in the blockchain network, the block is added to the blockchain. The miner gets a transaction fee and a new block addition fee in the case of PoW [5]. The ledger runs on a peer-to-peer network and thus all the nodes participating in the network get a copy of the original information.

1.2.1. Characteristics

As based on a peer-to-peer network. If any node's information in blockchain gets tampered with, it will not match the information copy on other nodes. As a result, the tampered copy will get discarded as the majority will not agree on the tampered copy to be true. Hence, any third party or broker is discarded by building a secured trusted

peer-to-peer network based on rules implemented by consensus mechanisms.

A blockchain network has the following key characteristics:

1) **Consensus:** All participants must agree on the validity of a transaction for it to be valid.
2) **Provenance:** Participants know where the asset came from and how its ownership has changed over time.
3) **Immutability:** No participant can tamper with a transaction after it's been recorded to the ledger [6],[7]. If there is an error in a transaction, a new transaction must be used to reverse the error. Both these transactions are then visible [8].
4) **Finality:** A single, shared ledger provides one place to go to determine the ownership of an asset or the completion of a transaction. The privacy of the Blockchain is maintained by high-end cryptographic hash functions and public-key cryptography. It also helps to achieve transparency.
5) **Smart Contract:** Smart contracts [9] are programs of a computer that support the transmission of money or anything of value. When a particular policy is met, these programs run automatically. Each smart contract contains a contract address, predefined functions, and private storage. Ethereum [10], is an open-source and decentralized platform that executes smart contracts. To construct a smart contract Ethereum platform uses solidity as a programming language.

1.2.2. Architecture

As discussed earlier, the initial block in a blockchain is known as the genesis block [11]. A genesis block does not contain a previous hash but the block's current hash. The **Figure 1.1.** shows a general view of a blockchain. A block contains the

transactions, the hash of a previous block, and a hash of the next block [1]. This information is stored in a block using a cryptographic mechanism. There are also some other fields in the block header which are shown in **Figure 1.2.** We explain each of them.

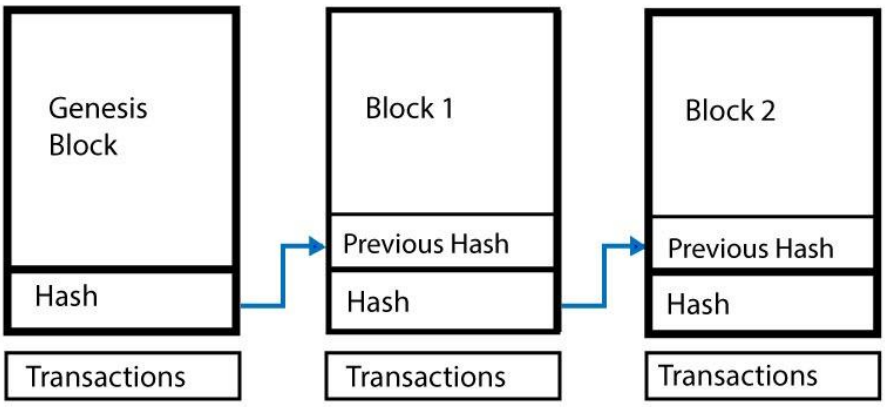

Figure 1.1. Blocks in Blockchain

1) **Previous Hash:** This field contains a hash of the previous block which connects the current block with its parent in the chain. This hash value is calculated from Secure Hash Algorithm (SHA) 256 in bitcoin. SHA 256 is a type of 'signature' for data where it generates a unique 256-bit (32-byte) signature for a text.
2) **Timestamp:** It is the time when a new block is created [12].
3) **Tx–Root:** This field is also known as the Merkle root, and it contains the hash value of the block which has all the validated transactions [13]. As shown in **Figure 1.2** a hash value is calculated out of transactions where these transactions are combined pair by pair and are then combined for another hash function. These steps are repeated until they all are combined in a single entity called Merkle root.

4) **Version:** This contains the version of the protocol used by that particular node that proposes a block to the chain.
5) **Nonce:** This field is used to prove the efforts that a node has paid for getting the right to append its block to the chain. This field will be presented in the next section.
6) **Bits:** This field represents the difficulty level of the PoW [14].

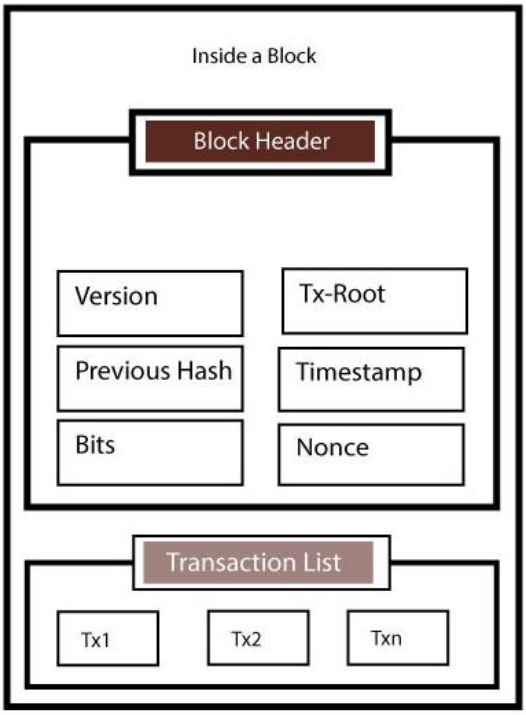

Figure 1.2. Structure of a Block

1.3. Privacy, Security and Pseudonymity

Privacy and Security is the essential right of each resident and must be integrated into every system. Privacy ensures the real identity of an individual/vehicle/node while security ensures properties such as confidentiality, authentication, availability, and integrity. To ensure privacy is maintained pseudonyms are given to hide the real identity of the entity. Pseudonyms are fake identities used instead of original names. The new principles depict the utilization of changing, pseudonymous identifiers rather than static identities. In ITS, as different remote systems, some attacks endanger ITS-S

(station/vehicle) privacy and security. An attacker can utilize bogus data and may gather messages of vehicles or track the location of vehicles [15], hence accessing user private data. The purpose of the discussion in this section is to give an insight into the importance of using pseudonyms where original identity is hidden to avoid different attacks.

1.4. Problem Statement

To achieve privacy and pseudonymity, PoW imposed heavy computational power and expensive resources. Another consensus is suggested for pseudonym management over blockchain but they need to be tested and compared. The proofs such as PoW, Proof of Elapsed Time (PoET), and Proof of Kernel Work (PoKW) for privacy and pseudonymity are not present in the algorithmic form to provide the implementation details. The suggested proofs are not compared with each other in terms of efficiency in pseudonym distribution schemes. This thesis compares the said consensus. The thesis also addresses curious Road Side Unit (RSU) that tries to analyze the communication habits between different real vehicles and the user's real identity behind the vehicle. External attackers can attack RSU by stealing the data stored in RSU. An attacker may forge a large amount of communication data within RSU and may perform other illegal operations. Hence RSUs are at risk and there is a need to secure RSU by deploying blockchain. We address all the above issues in our thesis.

1.5. Aims and Objectives

The aims and objectives of the research study are as follows

1. To develop a pseudonym-based shuffle mechanism to achieve privacy.
2. To evaluate various consensus algorithms designed for ITS pseudonym shuffling and designing a hybrid consensus algorithm i.e Proof of Pseudonym to tackle the shuffling mechanism efficiently.

3. To enhance the security of RSU in terms of protecting users' privacy and not to make it a single point of failure.

1.6. Proposed Solution

The proposed solution introduces an architecture based on a traditional ITS structure where a hierarchy of Public Key Infrastructure (PKI), Privacy Managers (PMs), Road Side Units (RSUs), and vehicles interact in the network. The PKI is used for the initial registration of the vehicles where it issues the original as well as pseudonym identities. Vehicles after using the pseudonyms give them to RSU which are then collected by PMs. PMs maintain a Blockchain in the cloud where it shuffles the used pseudonyms. These shuffled sets are then allotted to RSU where it distributes to vehicles. RSUs manages another blockchain that is used to secure the pseudonyms sets on RSU. We proposed a consensus method for the pseudonym shuffling scheme i.e. Proof of Pseudonym which is tested for the efficiency in the scheme. The results show that the proposed consensus achieves better efficiency in terms of execution time and memory it takes for an agreement.

1.7. Motivation

In current years, blockchain technology has a big aim in the field of distributed systems. It is a decentralized system that associates digital signatures, cryptography, time sequence, and hash functions. The technology of blockchain is based on unifying a ledger between all nodes in the network and ensuring that no external node can join the network without authorization [6]. In the case of vehicular ad hoc networks, blockchain can be used to solve the problem of the centralized database and make all the entities in the network be a part of the ITS management. And at the same time ensuring security and privacy in the networks. The purpose of this research is to

- Optimally deploy blockchain technology into the privacy protection of ITSs.
- The scheme achieves pseudonymity by shuffling the pseudonyms over a blockchain and achieving efficiency by improving the consensus mechanism.
- The pseudonym management is done through a central party that is Certificate Authority which is prone to attacks and is considered as a single point of failure. Blockchain eliminates the need for a trusted central party.

1.8. Significance

The significance of the proposed work is providing an optimal solution for pseudonym management via an efficient consensus mechanism for pseudonym shuffling. External attackers and curious RSUs cannot take an advantage of data kept in BC as the transactions at RSU are anonymous. Aims to improve the unpredictability of pseudonym mixtures. It also reduces the cost and effort of constantly generating new pseudonym certificates by shuffling used pseudonyms. The social advantage of blockchain is in terms of convenient traveling where users are recommended to take part in the blockchain network to share traffic and navigation information yet not disclose their identities. Through blockchain traffic routes can be optimized and speed can be controlled hence reducing pollution of urban traffic.

1.9. Thesis Structure

Chapters in this thesis are categorized as follows.

Chapter 2. Literature Review

In this chapter, we investigated the literature review on different methodologies for achieving privacy and security in ITS and evolving blockchain technology in detail. However, the main focus is on the consensus methods used in blockchain technology. We also discussed the use of blockchain technology in various applications. While

studying literature, we witnessed that the traditional consensus methods used in blockchain face some issues for real-time applications particularly. Moreover, there is a need for the development of a consensus algorithm that can efficiently tackle the shuffling mechanism discussed by previous researchers. Additionally, the existing mechanism for the shuffling scheme does not give attention to the security breaches of Road Side Units (RSU).

Chapter 3. Proposed Architecture

In the next chapter, we will present the proposed BC-based pseudonym management scheme for ITS. It is a distributed BC-based framework for tackling the pseudonym scheme efficiently. The architecture also gives insight for protecting RSUs from insider and outsider attacks.

Chapter 4. Implementation and Security analysis

This chapter contains the security analysis, implementation details, comparisons among proposed frameworks, and consensus methods in the literature. We used Node.js to implement the proposed framework. During implementation, various experiments are accomplished to analyze the time complexity of the algorithm, the running time of the proposed algorithm, memory utilization, and the overhead ratio of the proposed framework.

Chapter 6. Conclusion and Future Work

This chapter contains the conclusion, limitations of the research study and some future directions.

1.10. Summary

This chapter contains basic knowledge about Blockchain, ITS, Privacy, and Pseudonymity. Additionally, it also contains a problem statement and a summary of the proposed architecture. Furthermore, the chapter contains the aims & objectives of this research study. The end of this chapter contains the thesis organization. In the following chapter, we analyze the literature review on different schemes for achieving privacy in ITS and emerging blockchain technology in detail.

Chapter 2 Literature Review

2.1. Blockchain Technology

The 21st century is all about revolutionizing technology. One of the leading technologies that have turned many aspects is blockchain. It miraculously impacted different businesses from the very first step. Blockchain provides decentralized, transparent, and secure systems. It is a distributed ledger technology that maintains transaction ledgers and secures them by using cryptography. The transactions are recorded in blocks and these blocks are connected through hashes. Initially, it was used by Satoshi Nakamoto in 2008 for public transactions of bitcoins. Bitcoin [16] digital currency was the first application of blockchain [17, 18]. Blockchain came as a solution to the long-standing user's trust problem. With its emergence with the renowned cryptocurrency Bitcoin. It provided an architecture to allow the user to trust a decentralized system instead of trusting a third party. Operating top of a peer-to-peer network, it keeps records of the ledger of transactions. This helps to avoid any center party. The whole process is done through a consensus. A ledger is shared between multiple entities, allowing everyone to inspect. No single user can control it. It is a distributed cryptographically secured database that keeps the record of every transaction from the very initial one.

2.1.1. Features of Blockchain:

The following are the main features of Blockchain.

- **Decentralized Computation:**

Blockchain consists of distributed ledgers maintained by peer-to-peer networks. Blockchain eliminates the role of the central entity by using consensus protocol to validate transactions.

- **Distributed Ledger of Transactions:**

A shared ledger is used to store transactions[19, 20]. A copy of the ledger is maintained on every peer of the blockchain network. These copies are synchronized by timely replication [21].

- **Transparency:**

Blockchain stores every transaction in a block. Also, it is available to all peers for verification [22].

- **Security:**

Each block is added to the chain after validation. Also, each block contains a hash of the previous block [23]. It is computationally impossible to delete or update a block because it requires re-calculation of all the preceding blocks' hash.

- **Fault-Tolerant Network:**

Blockchain has a peer-to-peer network of nodes. All miner nodes process transactions in parallel [24]. The blockchain will continue working if some of the nodes fail to function.

2.1.2. Types of Blockchain

There are three main types of blockchains. These do not often confuse traditional databases or distributed ledger technology (DLT) with blockchains. There are three types of blockchains discussed as follows.

i. Public/Permissionless Blockchains

The blockchain that has no restriction on accommodating anonymous participants is known as permissionless blockchain [25]. The term public blockchain is used

interchangeably. Lottery-based consensus algorithms are used to publish a block. A single node handles publishing a block. If voting-based consensus is allowed to be used in permissionless blockchain, multiple accounts can be made by the participants to do a Sybil attack to make the decisions in their favor. A Sybil attack is one of the issues in peer-to-peer networks where a malicious node creates many identities and tries to manipulate the network by controlling it. Public blockchains need security and for this purpose, the block creation mechanism needs to be difficult and expensive so that the resources of one node are not enough to bias the decisions in its favor. There are various consensus algorithms for private and public blockchains which are discussed in this chapter.

ii. Private/Permissioned Blockchains

Private and consortium blockchains are permissioned blockchains. In such types of blockchains, the number of participants is limited and they keep a copy of the blockchain [26]. The consensus mechanism is not much expensive for publishing a new block. All the participants in the permissioned blockchain are known so the risk of Sybil attack is eliminated. For the consensus, voting mechanisms are used. Hence permissioned blockchains have higher performance than permissionless blockchains. Non-public blockchains are divided into fully private or consortium blockchains. An organization can choose one of them based on the cost and needs. A consortium can be the best option if organizations want to share costs and data.

iii. Hybrid Blockchains

A hybrid blockchains combine the privacy of a private blockchain with the security and transparency of a public blockchain [27]. This gives the businesses a significant amount of options to choose from what they want to keep private and what to be made public. For example, Dragonchain blockchain is a hybrid blockchain [28]. It allows its

users to connect with other blockchain protocols. Thus, allowing blockchains multichain network. Being able to post to multiple public blockchains at once increases the security of transactions, as they benefit from the combined hash power being applied to the public chains [29].

2.1.3. Applications of Blockchain

Blockchain has now been deployed in not only cryptocurrency but its underlying technology is used in various applications [30]. We tried to discuss a few applications of blockchain which includes cryptocurrencies as well as other potential areas where blockchain has emerged. **Figure 2.1.** shows different applications of blockchain. However, the applications are not limited to the ones discussed in this thesis.

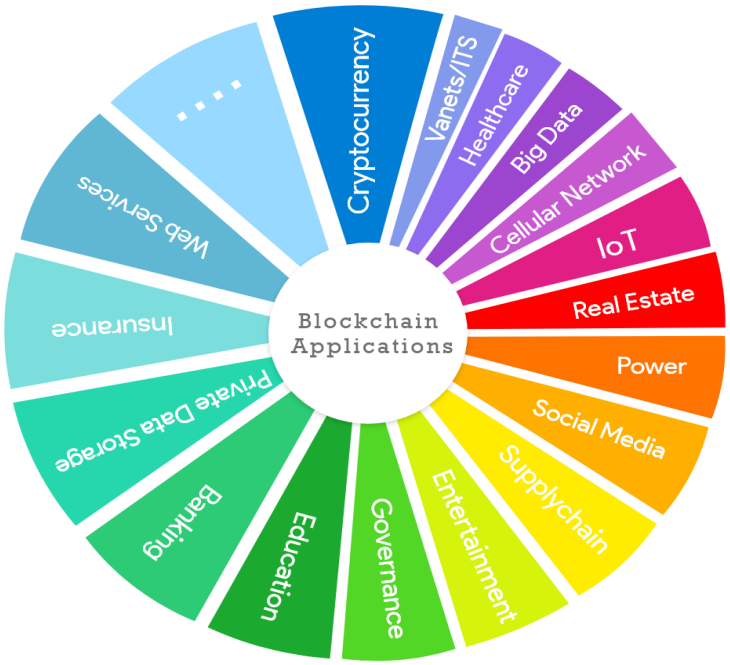

Figure 2.1. Blockchain Applications

a) Cryptocurrency

Over the past decade, cryptocurrency is being an evolving topic, merging incredible technical power and enticing investments worth trillions of dollars on a worldwide scale. The underlying technology of cryptocurrency is attractive for many other

applications due to its unique features and architecture. This is why it is becoming popular due to its applicability, efficiency, and data-centric characteristics. Cryptocurrencies such as bitcoin use blockchain technology to secure transactions using blockchain with the Proof of Work algorithm.

b) Private Data Storage

The transactions are used to carry instructions for queuing, storing, and sharing data. As the increased number of mobile applications need access to complete access of user data that is: contacts, photos, messages, and other important user data. Zyskind et al. [31] proposed blockchain implementation with other offline storage methods to provide permissions on each set of data. LevelDB or cloud storage can be used to bound the data on the blockchain. This could lead to a little dependency on third parties but provide a more scalable solution. Companies can go for upgrading technology find more reliable solutions for the data that needs privacy solutions.

c) Education

Sharples and Dommingue [32] suggested the use of blockchain to keep educational records and rewards. They also suggested the use of educational reputation currency to be given as a reward. How the founder can use blockchain in education for online courses. This technology can record the student signed up for the course and verify that the student has completed and learned the course. A payment feature can be added for the use of smart contracts by students to ensure lifelong learning plans.

d) Banking

Blockchain can be useful in the computerization of various niche aspects of banking like data loss reporting, client account reconciliations, clearing settlement and Over the Contracts (OTC) contracts/products, etc. Classic banking methods like an

endorsement of a loan or derivative is a time taking process due to several back-end stages which involve contract consultations with multiple parties. Blockchain provides the essential transparency and speed through smart contracts, to this necessity. Various banks are already testing Blockchain and they are getting services from technology companies such as IBM, R3, and Microsoft.

e) Voting

In the year 2014, a Danish political party was the first to use blockchain technology for voting [33]. 'Followmyvote' offers an online voting platform that follows blockchain technology for a secure voting system [34]. A challenge for a fair voting system that keeps users vote privacy and that provides transparency and flexibility of the electronic systems is solved in this paper. A novel blockchain application for fair electronic voting is proposed by [35] which eliminates some of the issues of the existing system. It particularly addresses the election process which reduces the cost of presenting elections nationwide.

f) Blockchain in Internet of Things

IoT is a network of connected vehicles, home appliances, physical devices, and other items that are accessible through the Internet [36]. IoT is widely used in smart homes, smart grids, intelligent manufacturing, intelligent transportation system, and other fields [37]. The traditional centralized does not guarantee trusted interaction among devices and the security of sensitive information. Therefore, the combination of blockchain and IoT becomes an expected trend, where smart contracts will help to automate, promote resource sharing, complex workflow, ensure safety, efficiency and save costs [38, 39].

g) Blockchain in Social Media

Blockchain has played an important role in social media applications. The reason it's spreading and being liked is privacy. Being on a blockchain takes away the concept of being centralized. The most common examples are Ushare [40], DUST, BeeChat, etc. Some of those applications provide anonymity which may give way to malicious behavior. We now discuss some of the top blockchain-based social media networks as shown in **Figure 2.2.**

i. Steemit

Steemit provides the services of both Redit and Facebook and so it's one of the top blockchain-based social media apps. This platform allows users to post their pictures,

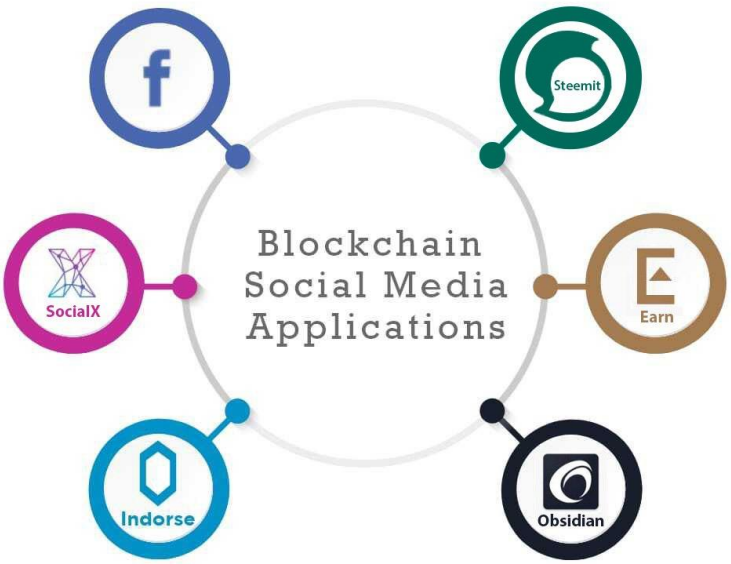

Figure 2.2. Blockchain in Social Media

posts, music, and video and get paid for the content. This platform automatically distributes Steemit currency units to active users.

ii. Earn

Earn is better than LinkedIn. Earn allows users to get earned by completing microtasks. Users need to create a profile and they would get paid messages or they

would get an opportunity to join people with similar skills. The paid email system of Earn can increase the chances of making money.

iii. SocialX

SocialX is very much similar to Instagram and Facebook except that it's decentralized. This platform has all the social network features and it allows users to post pictures and videos on a highly secure platform. SocialX has built-in license management which gives users the choice to sell their photos and videos or keep them to themselves.

iv. Obsidian

Obsidian provides a wide collection of blockchain-based apps but the main service is a secure messenger. This allows users to chat 100 \% privately and securely. Obsidian provides end-to-end encryption which proves the files and messages are seen only by the intended recipient.

v. Indorse

Indorse is a professional platform based on ethereum blockchain. This is a LinkedIn equivalent which increases the chances for users to get profit for their skills. Users get paid or get impressive rewards for their contributions. For example, if a user is good at coding he will be validated by experts or a user can get a good job by showing business skills.

h) Blockchain in Power

Another area of blockchain is to enable customers to switch power suppliers quickly. Companies are also using blockchain for meter registration to make the process less costly and more efficient. Blockchain may also make existing electric industry methods more efficient by helping the utilities' "smart grid" management systems that

spontaneously diagnose network problems and emergencies and in reaction reconfigure the network [41].

2.1.4. Consensus Mechanism

We know that a blockchain is a decentralized distributed network that provides security, immutability, transparency and privacy. There is no concept of centralization to verify and validate the transactions, but still, transactions in the blockchain are considered to be completely verified and secured. This is the result of a core algorithm present in every blockchain network called consensus protocol. A consensus algorithm is a technique through which all the peers of the blockchain network reach a common agreement about the current state of the distributed ledger. So consensus algorithms provide trust and reliability among unknown peers in a distributed environment. The consensus mechanism ensures that every new block added to the blockchain is the only truth that is agreed upon by all the blockchain nodes [42]. **Figure 2.3.** shows a categorical diagram of the consensus and their distribution.

a) Consensus Algorithms used by Various Cryptocurrencies

Digital currencies have gained faster payment methods by using blockchain. Bitcoin was the first cryptocurrency used by a blockchain. After gaining popularity in bitcoin other applications were/are implemented. **Table 2.1.** defines some of the important characteristics of consensus algorithms used in cryptocurrencies.

i. Proof of Work

Proof of Work (PoW) [43] is the first consensus protocol used by a public blockchain. All the nodes needed to solve a cryptographic puzzle by brute force. The node which wins the puzzle is rewarded and all the other nodes computations are wasted. The consensus is achieved as 51% of the power.

ii. Delayed Proof of Work

This protocol is designed as a hybrid consensus in which one blockchain takes security from another blockchain through hashing power [44]. A group of nodes is responsible for adding data from the first blockchain onto the second. Both blockchains would then compromise to undermine the security of the first blockchain. Komodo which is attached to the bitcoin blockchain was the first one to make use of this protocol.

iii. Proof of Stake

Original Proof of Stake (PoS) uses the wealth of miners to win a ticket rather than computational power. PoS was first implemented in 2012 as cryptocurrency PeerCoin. PoS is kind of a hybrid design where PoW is used in the beginning for coin minting and later PoS is used for the security of the whole network. PoS works with the concept of coin's age which is explained by example. If 10 coins are held for 10 days its age is 100 days. If these coins are spent, their age is consumed. In PoW the mainchain with most work is followed, and in PoS a chain with most coin age is followed. Cardano's Ouroboros, the version of PoS adds security measures to ensure persistence and liveness within the system. This implementation elects the stakeholders through a delegation process and takes the snapshots of current stakeholders labeled as an 'epoch'. The subset of current stakeholders randomly decides who will be the next epoch stakeholder.

iv. Stellar Consensus Protocol

Stellar Consensus Protocol (SCP) is a decentralized consensus protocol in which nodes can choose which nodes to trust. This group of trusted nodes is known as the "quorum slice". An agreement is reached by a set of nodes called quorum whereas a quorum slice is a subset of a quorum that selects one particular node for an agreement.

SCP proposes new candidate values for agreement via a "nomination protocol". Each node will then vote for a single value among these. After this, a "ballot protocol" is implemented. In this phase, the nodes vote for the previous values to keep or abort them. In case the quorum slice doesn't reach consensus the value is shifted to a higher valued ballot so that it can be voted on again.

v. Delegated Proof of Stake

Another form of Proof of Stake (PoS) is Delegated Proof of Stake (DPoS). It is the same as PoS except the stakeholders elect their delegates to generate and validate the blocks. As there are fewer nodes to validate, blocks can be validated quickly and the transactions can be confirmed quickly. Delegates can tune the block size and intervals in the meanwhile [45].

vi. Leased Proof of Stake

Leased Proof of Stake (LPoS) allows the nodes with a low balance to participate in solving the blocks [46]. The nodes with low balance take some amount on lease from the nodes with high balance. The amount is in the control of the wealthy owner. When a block is solved by these nodes the reward is shared with the wealth holders. This approach is more decentralized hence making the blockchain more secure.

vii. Byzantine Fault Tolerance

When nodes can generate arbitrary data, i. Byzantine Fault Tolerance (BFT) is a replication algorithm that can solve the issue of reaching consensus [47]. BFT can guarantee the aliveness and safety of a system. It can tolerate up to 33% of faulty nodes [48].

viii. Directed Acyclic Graph

In Directed Acyclic Graph (DAG) the data is stored topologically in a graph manner. DAG can overcome the problems of data processing, compression, and routing. One of the disadvantages of PoW is the creation time of the block which is 10 minutes. DAG instead of implementing a single chain, works on "side chains" [49]. So to reduce the time of block creation and validation different transactions are performed on multiple chains. Mining also is a waste of time and energy so in DAG all the transactions are maintained and directed in a certain sequence. DAG is acyclic so there is no chance of finding the parent node as it's a tree of nodes and not the loop of nodes.

ix. Proof of Weight

This is a very good alternative to PoS. In PoS if the participant has more tokens, he is the winner of the block but this idea makes it a bit biased [50]. So to solve this problem of biased PoWeight was introduced. It works on other factors than on tokens. Cryptocurrencies like Filecoin, Algorand, and chia implement this algorithm. These factors are known as "Weighted Factors". In Filecoin the IPFS data that a system has is the weighted factor. There are also some other factors like Proof of Reputation and Proof of Spacetime. Proof of Weight system provides scalability and customization however incentivizing can be a big challenge for this algorithm.

x. Delegated Byzantine Fault Tolerance

Delegated Byzantine Fault Tolerance (DBFT) algorithm is a different version of BFT. This fault-tolerance algorithm divides P2P into two types' ordinary nodes and bookkeepers. Bookkeepers are elected by ordinary nodes who vote for the bookkeepers to take part in the consensus process. A random bookkeeper broadcasts its transaction to the network and 66% of bookkeepers should agree on the validation of transaction data. Upon validation, the transaction is appended to the blockchain. For

another consensus process, another bookkeeper is selected via the same process as discussed in [50].

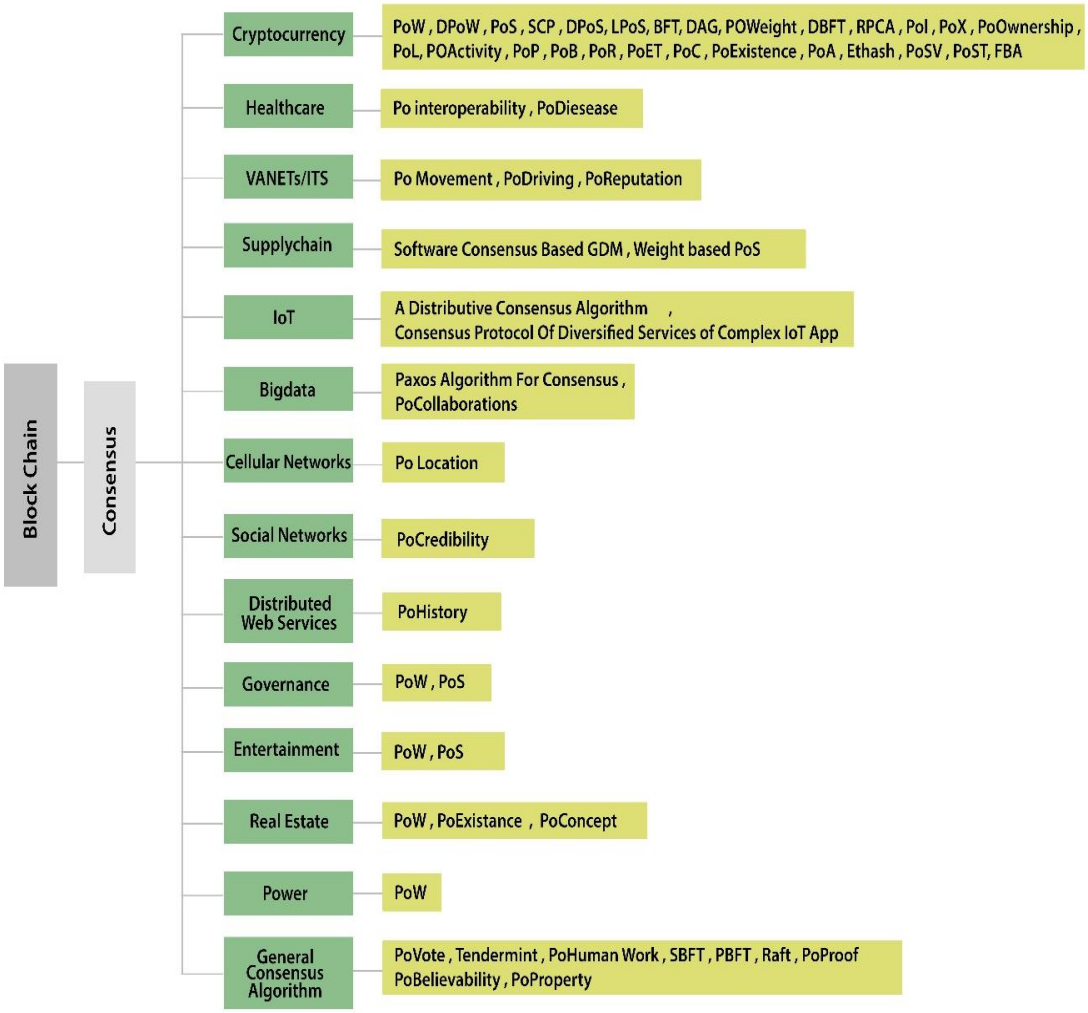

Figure 2.3. Categorization of the Consensus Algorithms

Table 2.1. Characteristics of Cryptocurrency Consensus Algorithms

S.No	Consensus Algorithms	Permissioned / Permissionless	Platform	Programming Language	Advantages	Disadvantages
1	Proof of Work (PoW) [41]	Permissionless	Bitcoin	C++	Better security, Suitable for a variety of applications	Wastes considerable energy, 51% Attack possible Advance hardware required

2	Delayed Proof of Work (DPoW) [42]	Permissionless (But it can be customized as permissioned)	Komodo	Python	Energy-efficient, Increased security	Blockchain using Pow and PoS can be part of this consensus
3	Proof of Stake (PoS) [51]	Permissionless	Peercoin.Nxt, Blackcoin, Shadow coin, Ethereum	Java	High speed Less energy consumption, No advanced hardware required	Rich get richer
4	Steller Consensus Protocol (SCP) [52]	Permissionless	Steller	C/C++, JavaScript, Go, etc	Fast transactions with low fees	Inefficient in case of several messages sent
5	Delegated Proof of Stake (DPoS) [43]	Permissioned /Permissionless	Steem.it, EOS, Bitshares, Lisk	Javascript, C++, Ark	Secure Real-time Voting, Better distribution of rewards	Cartel formation, Easier for 51% attack, Partially decentralized
6	Leased Proof of Stake (LPoS) [44]	Permissioned	Waves	Scala	Fair usage, lease coins	Decentralization issue
7	Byzantine Fault Tolerance (BFT) [52]	Permissionless	Ripple, Steller, Hyperledger Fabric	Not known	Less Energy consumption, No advanced hardware, Fast, Scalable	Less suitable for public blockchain
8	Directed Acyclic Graph (DAG) [50]	Permissionless	Iota, Hashgraph, Byteball, Raiblocks/Nano	Javascript, Rust, Java, Go, C++	Low-cost Network, Scalable	Difficult implementation, Not good for smart contracts
9	Proof of Weight [50]	Not Known	Filecoin, Algorand, Chia	SNARK/STARK	Scalable, Customizable	Incentivization issue
10	Delegated Byzantine Fault Tolerance (DBFT) [50]	Permissioned/ Permissionless	Neo	C#, Python, .Net, Java, C++, C, Go, Kotlin, Javascript	Fast, scalable	Conflictions on the chain
11	Ripple Protocol Consensus Algorithm (RPCA) [52]	Permissionless	XRP	Java, C++, Node.js	Energy efficient, quick	Centralization
12	Proof of Importance (PoI) [50]	Permissionless	XEM	Java, C++	Vesting Transaction partnership	Decentralization issue
13	Proof of Exercise (PoE) [52]	Permissionless	NA	NA	Avoid wastage of computational power	Needs dedicated research for practical implementation.
14	Proof of Ownership (PoO) [53]	Permissionless	Decred	Go	The use of unique pseudonyms makes multiple attacks difficult	Not known
15	Proof of Luck (PoL) [53]	Not Known	TEE (Trusted Execution Environment) such as Intel SGX-enabled CPUs	Not known	Decentralized, Low latency with the transaction validation, Power safe	Prone to revision attack Forking, Unfair
16	Proof of Activity (PoA)[50]	Permissionless	Bitcoin or Bitcoin-related technologies	Solidity, Java, Python	Equal contribution, Reduces 51% attack	Better energy consumption, Double signing

17	Proof of Publication (PoP) [51]	Permissioned	Bitcoin or Bitcoin-related technologies and General Applications	Python, C++, Shell, Javascript	Can be used in cryptocurrency and general applications	No Energy saving
18	Proof of Burn (PoB) [51]	Permissionless	Slimcoin/Redcoin	Golang, C++, Solidity, LLL Serpent	Network preservation	Coins wastage, Not good for short term investors
19	Proof of Retrievability (PoR) [54]	Permissioned/ Permissionless	Microsoft, Permacoin	Golang, C++, Solidity, LLL Serpent	Efficient	Extending the number of queries is challenging
20	Proof of Elapsed Time (PoET) [55]	Permissioned/ Permissionless	Hyperledger sawtooth	Python, Javascript, Go, C++, Java, and Rust	Cheap participation	Special hardware, Not suitable for public blockchain
21	Proof of Capacity (PoC) [51]	Permissionless	Burstcoin, Chia and spacemint	Java	Efficient, Distributed, Cheap, Utilizes free disk space as a resource	Decentralization issue
22	Proof of Existence (PoE) [56]	Permissionless	Poex.io, Hero Node, Dragon Chain	Not known	Document time stamping, Document integrity	Not known
23	Proof of Authority (PoAuthority) [56, 57]	Permissioned	PoA. Network, Ethereum, Kovan testnet, vechain	Solidity, Java, Python	Reduced maintenance costs	Centralization
24	Ethash [58, 59]	Permissionless	Ethereum	Python, Go, Java, Javascript, Ruby, C++	Avoids 51% attack	Memory intensive, Needs computers with powerful GPUs
25	Proof of Stake Velocity (PoSV) [60]	Permissionless	Redcoin	Not known	Reduces the time wastage of mining, Removes mining arms race	Not known
26	Proof of Space-Time (PoST) [61]	Not Known	Filecoin	Go, Javascript	Cheap computationally	Needs more interaction
27	Federated Byzantine Agreement (FBA) [62]	Permissioned/ Permissionless	Steller and Ripple	C/C++, Javascript, Go, Java, Node.js	Fewer participants to achieve consensus' Robust	The parties must accept the exact number of candidates

xi. Ripple Protocol Consensus Algorithm

Ripple cryptocurrency uses (RPCA) and it was designed to address other algorithms latency issues. RPCA works as follows: Each server puts all the valid transactions in the "candidate set" which is the public list. Each server gathers all candidate sets, from other Ripple servers which are found in its unique node list. Every server then votes for the validity of transa

ctions. This voting can be done in one or multiple rounds. A minimum of 80% yes is required for all the transactions in the final round to be written into the public ledger and then the ledger is closed [63].

xii. Proof of Importance

Proof of Importance (PoI) is used by the cryptocurrency XEM which is used by NEM [50]. Every account has a vested and unvested XEM balance. Unvested balance is the amount received. After every 1440 blocks, one-tenth of the unvested balance goes into a vested account. XEM is spent from both vested and unvested accounts when a XEM needs to be sent. This is so because both accounts need to retain the same ratio.10, 000 XEM is the minimum amount an account should hold in its vested part to be eligible for "Importance Calculation". Importance is calculated on a weighting factor i.e to check if an account is a part of cluster nodes or an outlier, on the amount of vested XEM, the rank of the account within the network. The ranking is held via the NCDawareRank algorithm and the NEM network determined two suitable constants.

xiii. Proof of Exercise

Proof of Exercise (PoX) uses computational power for scientific problems. In Proof of Exercise, the employers give the matrix-based problems to miners. There are two reasons for using matrices: tuning of the network difficulty becomes easy and they are a principal abstraction for many scientific problems. A "hostage credit" system is placed in the system so that the data for matrices required is readily available. Miners need to bid for a problem to solve and deposit which will be refunded after successful completion of the problem. To avoid complicity among miners, verifiers, and employers the matrix problem is sent via shuffling service. This is done either directly after the miner sends the data for verification or directly after the matrix data is being

published by an employer. In addition, if the bid is won by multiple miners at the same time the coin reward is shared.

xiv. Proof of Ownership

Proof of Work is prone to a Sybil attack where an attacker can do the amount of work multiple times as he acts as multiple participants. Using a TEE a participant needs to own a unique CPU instead of virtually maintaining participants. EPID signature is used in Proof of Ownership (PoO) protocol which produces pseudonyms that show if the multiple proofs are coming from the same CPU. The PoO generates unique pseudonyms so that if a malicious user resets the owner epoch register for using it multiple times, the attacker may not be able to do that. So a consensus in the blockchain is reached by following a block having most proofs with unique pseudonyms.

xv. Proof of Luck

Proof of Luck (PoL) aims to increase transaction throughput and reduce the computational power used by PoW. Each block when mined is given a random number between 0 and 1 which is called a "luck value". Higher numbers are considered as luckier and less unlucky. The highest luck value is calculated by adding all the values of each block starting from the genesis block till the last block. The miners prefer to append their block to the blockchain having the highest luck value. A higher luck value produces less delay time and optimizes communication within the system. The original miner will not need to broadcast its block to the network as another miner tries to solve the proof on the first block having a higher luck value [53].

xvi. Proof of Activity

There are many disadvantages of Proof of Stake which include keeping coins for long with oneself. Coins are also sent to the transactions which further assigns coins to the rest destroying the coinage but they are not included in PoS. When the node is offline, coins are still collected and this is the main weakness of PoS. When the node is occasionally online, there is a delay in receiving their incentive which results in incentive distributions bursts. If the number of online nodes is insufficient it may result in attacks. PoA is a combination of Proof-of-Work (PoW) and Proof-of-Stake (PoS). PoA protocol offers good security against possibly practical attacks on PoW and has a relatively low disadvantage in terms of network communication and storage space. The likelihood of a 51% attack on a PoA system is reduced considerably. This is because a mischievous actor would need majority control of both the number of coins in a system and the hash rate of mining.

xvii. Proof of Publication

Blockchain provides hashes that are linked however the servers may backdate the records by hashing and signing the past timestamps. To handle this issue the timestamps are linked. This technique takes the timestamp of the digital record's creation and modification takes a hash out of record and timestamps and links them together. So even if the clocks are incorrect, this technique can give surety of the complete order of records.

xviii. Proof of Burn

Proof of Burn (PoB) is also known as POW without energy waste. The miners instead of using heavy power utilize virtual token coins to burn and destroy to get a right to write into the blockchain. The miners buy mining rigs which give them the power to

mine blocks. Miners can send and get transactions by burning their own and other coins respectively. More coins burnt result in more virtual mining rigs.

xix. Proof of Retrievability

Proof of Retrievability (PoR) provides mining resources with the additional ability of distributed storage of archival data. It is similar to PoB but it involves not just computational power but also storage. This consensus is more suitable for cloud computing where a file system (Prover) can give surety to a client (verifier) that the file is intact. PoR is well known for permacoin and koppercoin [54].

xx. Proof of Elapsed Time

In Proof of Elapsed Time (PoET) each node is given a randomized timer object from a trusted code. The node having the shortest timer is when expired the node wakes up, propagates a signed certificate to show this node is the block leader. The timer is given randomly so that the malicious user does not try to continuously get the shortest timer [55].

xxi. Proof of Capacity

Proof of Capacity (PoC) is also known as Proof of Space. Instead of using miners' computational power, mining devices' storage capacity is used to store the possible solutions for mining crypto coins. More storage space results in higher chances of winning the mining reward. The list of solutions is stored on the device hard drive before the mining process begins.

xxii. Proof of Existence

Traditional models of validating the documents are based on central authorities which can lead to security breaches. Through blockchain Proof of Existence (PoE), the document can be stored with signature and timestamp associated with a legal

document [64]. The user can validate the document anytime. This is also advantageous as the blockchain is not centralized so the user can get privacy and security by having the proof of the document decentralized where it cannot be modified by a third party.

xxiii. Proof of Authority

Proof of Authority (PoA) was proposed for private networks as part of the Ethereum ecosystem. In PoA authority is given to N nodes. Each node is given a unique id. These authorities are responsible for running the consensus and ordering the clients issued transactions. PoA runs on a mining rotation schema in which the responsibility of block creation is distributed fairly among the authorities [57].

xxiv. Ethash

Ethash is a memory-intensive proof of work algorithm which is used by Ethereum [56]. In bitcoin, a block is created after every 15 seconds approximately as the difficulty level is adjusted automatically. However, Ethereum depends on the mining of 1GB data set, and these data sets are produced out of the headers of previous blocks after every epoch. i.e after about every 5.2 days or 30000 blocks. The clients of Ethereum store and generate the future data set in advance because the data set can take a long time to generate. This can prevent delays in mining at the beginning of every epoch. Ethereum designed Ethash with an aim of an "ASIC resistant" mechanism which reduced the advantage of joining mining pools. Since Blockchain mining centralization can introduce a risk of 51% attack. So Ethereum relies on memory instead of computational power. This also makes ASICs less attractive [58], as graphic cards of the top range are capable of mining Ether.

xxv. Proof of Stake Velocity

Proof of Stake Velocity (PoSV) is used in red coin cryptocurrency. It is a bit alternate to proof of work (PoW) and proof of stake (PoS). The ownership is referred to as stake and activity is referred to as velocity. It is based on the frequency at which a currency unit is used in the economy at a given time. The higher velocity is a sign of a better economy. Here is the formula to achieve proof of stake velocity: Vt =nT / M. Here 'Vt' is the currency unit's velocity, 'nT' is the aggregate for transactions while 'M' is the amount of money in circulation [59].

xxvi. Proofs of SpaceTime

In Proof of SpaceTime (PoST) a prover has to convince a verifier that data and space have been stored by him/her over some time. This data and space stored over some time are known as "spacetime" resources. PoST uses less energy as compared to proof of work as it requires that the difficulty level be increased by prolonging the period in which the data is stored rather than increasing computation costs [65].

xxvii. Federated Byzantine Agreement

Federated Byzantine Agreement (FBA) is well known for its low transaction costs, scalability, and high throughput. Steller and Ripple cryptocurrencies use this consensus where Steller was the first one to use FBA. It works like a byzantine fault tolerance where a blockchain is a responsibility of each byzantine general which belongs to the same blockchain. Nodes need to be known and verified in advance before the user requests any enactment from Federated Byzantine Agreement (FBA) [60]. The notary node selects those nodes who they trust, making quorums of nodes and hence forming the FBA network.

b) Healthcare Based Consensus

Blockchain is helpful in healthcare as it provides security to the medical records and other security breaches that healthcare faces daily. **Figure 2.4.** shows the patient profile over the blockchain which can be used for information by the concerned doctor, hospital, insurance company or it can be used in the supply chain management.

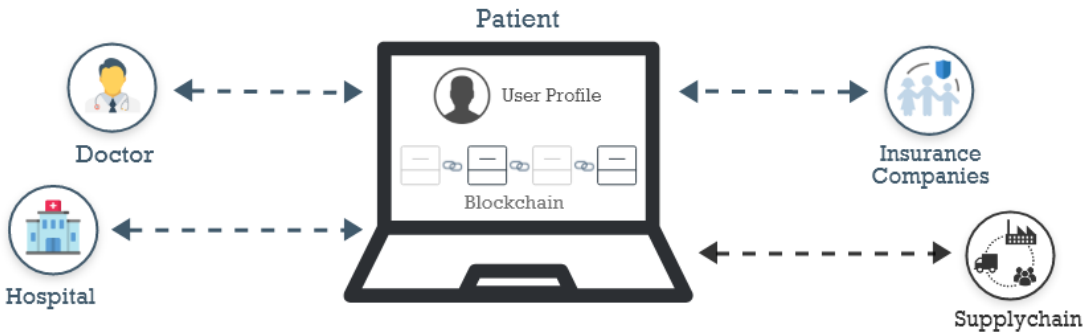

Figure 2.4. Blockchain in Healthcare

The consensus algorithms used in healthcare are discussed below.

i. Proof of Interoperability

Proof of Interoperability removes some of the disadvantages of Proof of Work. This is designed to achieve something fundamentally valuable. This protocol verifies that the incoming messages are interoperable with the known set of semantic and structural constraints. The use case discussed by Kevin Peterson and Rammohan Deeduvanu [62] in their research work is the FHIR profile. FHIR is the Fast Healthcare Interoperability Resources is an evolving standard that shows elements and data formats, along with providing publicly accessible Application Programming Interfaces (APIs) for the reason of exchanging Electronic Health Records. Proof of Interoperability requires a network to reach a consensus on the set of allowed FHIR profiles which includes the value sets of the attendant as well. This type of consensus

requires a human-based process to reach. Participants negotiate with the help of clinicians and terminology specialists. This consensus cannot, however, be reached programmatically. This collaboration demands a centralized repository. The value set repository proposed in this paper is the Value Set Authority Center (VSAC).

ii. Proof of Disease

There are several steps in POD which are as follows.

- Mobile devices and desktops are used as user devices and the server is a cloud-based application. Server and client communication is done via JavaScript Object Notation (JSON) objects.
- The patients enter their details of disease in simple English and the server runs hunspell using corpus (customized medical dictionary) over the user text.
- The user text is parsed using meta thesaurus and UMLS (Unified Medical Language System) and then the text is converted into multiple UMLS CUI (Concept Unique Identifier)
- The important information is either taken from the user online if the information is not available in EMR/HER (Electronic Medical Record/Health Electronic Record).
- The medical specialists called Medical Miner (MM), validate and confirm all the results from the above steps and commit them into the blockchain [66].

iii. Medical Information sharing using PBFT

As discussed by [67] PoW utilizes much energy and resources so instead of PoW they used PBFT in their application of blockchain for medical information sharing. PBFT can tolerate one-third of nodes to be malicious hence it is much more efficient in reaching consensus than PoW.

c) Intelligent Transportation System Consensus Algorithms

Recently smart vehicles have gained much attention in the area of research. The vehicular network is composed of various sensors, on-board units (OBU), and roadside units (RSU), etc where communication is exchanged among the nodes. Security issues may arise when an adversary tries to forge the message or tries to divert the traffic in the case of a platoon. Blockchain helps in securing the communication as all the communication is being done via transactions where they are recorded in a distributed ledger as shown in **Figure 2.5**. There are consensus algorithms discussed in different scenarios of VANETS and ITS.

i. Proof of Movement

Road miners (Smartphones or computers etc) share their transportation data with the community and get an automatic reward (Tokens) called zooz. These tokens can be used to pay for ride-sharing and other services. Road miners get more rewards if they drive for long (Incentive layer) [68].

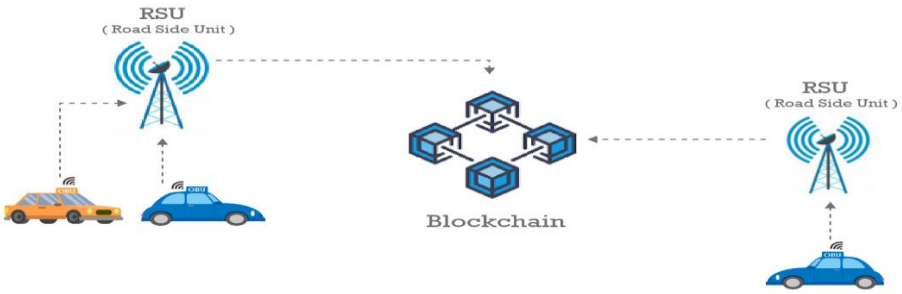

Figure 2.5. Blockchain in VANETs

ii. Proof of Driving

Proof of Driving (PoD) validates and verifies the vehicles in communication. IV-TP (Intelligent Vehicle Trusted Point) is the crypto data that is assigned to each vehicle and if a vehicle wins the consensus competition, it gets more IV-TP from the benefiter

IV. The vehicle having more IV-TP is leading the vehicle's communication network. This way it creates a trusted environment between vehicles communication [69].

iii. Proof of Reputation

The paper discussed so far in this section is about a decentralized reputation based on blockchain in vehicular networks. The received messages are rated by the elected vehicle from the crowd and then broadcasts its ratings which are in the form of blocks. Using the vehicle's local knowledge they validate the block and decide to add the block to the blockchain or not. Thus the rating which is stored on the blockchain is said to be reliable enough as these are validated by most of the vehicles in the network [70]. Proof of reputation can also generally be used for almost any business network. Gochain uses Proof of Reputation for Dapps and smart contracts that aim to decrease energy consumption, increase performance, provide network security and decentralization.

d) Consensus in Supply chain

A supply chain is a service of producing goods and products and delivering them to the ultimate customer. It deals with the manufacturers, suppliers, warehouses, organizations, retailers, and distribution centers where raw material is changed into fine deliverables. Blockchain has revolutionized the supply chain process drastically by reducing delays, costs, and human errors. All the changes made during the process of the supply chain are recorded in the transactions ledger keeping it secure and unchangeable. In **Figure 2.6.** the supply chain process is described where raw material is packed with RFID tag and barcode is used to finish the goods.

i. Soft consensus-based group decision-making

Acting in an isolation for the decision in the supply chain cannot be that helpful than making joint decisions of planning and execution using Supply chain coordination (SCC). A methodology named a fuzzy TOPSIS (Technique for Order Preference by Similarity to Ideal Solution) based MCDM (Multi-Criteria Decision Making) for selection problems of SCC is proposed in this paper. As the decision-makers are separated geographically, they give their preferences via the internet. A consensus should be reached for the preferences among the decision-makers and in this regard, this paper presents a soft consensus-based GDM (Group Decision Making) methodology. All the supply chain partners reach a consensus by forming a decision matrix.

This is an objective weight determination methodology that is used for the assessment of the weights of the criteria without the mediation of the decision-makers. This methodology is used as the meeting is internet-based [71].

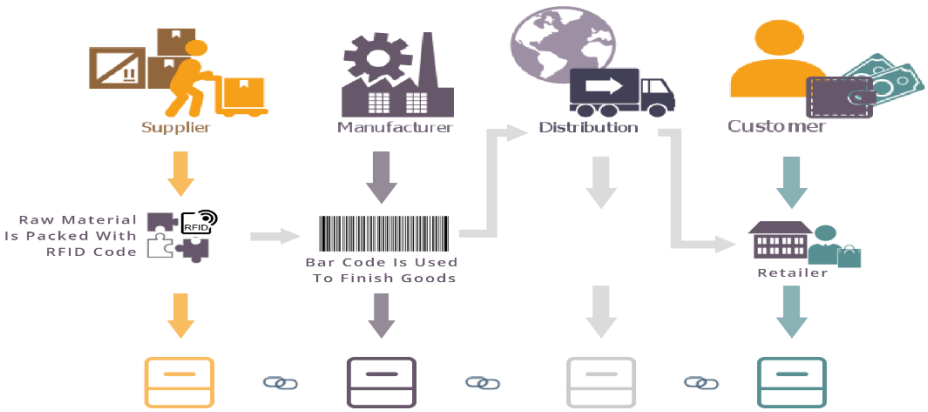

Figure 2.6. Blockchain in Supply chain

ii. Weight-based Proof of Stake

As discussed by Leng Kaijun [72] public blockchain consensus has a slow speed. The selective incentive weight should be used in common for the agricultural business resources so that these resources reach the underdeveloped and remote areas. Therefore this paper presents a consensus algorithm for the blockchain of agriculture business resources. The algorithm considers weight based on PoS.

e) Consensus in the Internet of Things

Internet of Things (IoT) relies on a centralized system where many devices are attached via the cloud or any other central system [73]. The data is sent back from the cloud to the device. This makes the scalability issue as sending and receiving of data from many devices can slow up the central system and security issues may arise. Blockchain has made the IoT more reliable, secure, and efficient. **Figure 2.7.** shows the scenario of blockchain retrieving data from various IoT devices. In different scenarios of IoT and blockchain different or already discussed consensus of the blockchain are discussed below.

i. A Distributed Consensus Algorithm

Global consensus might be the need to facilitate service integration and knowledge sharing. The idea of local consensus is developed and this is developed by each of the IoT edge nodes when needed [74]. Clusters are formed out of network nodes and each cluster reaches a local consensus. This local consensus can be used to make consensus decisions in the integration of functional capabilities and knowledge sharing. Any service can become part of the service pool where a local consensus is achieved for all the edge nodes involved in IoT.

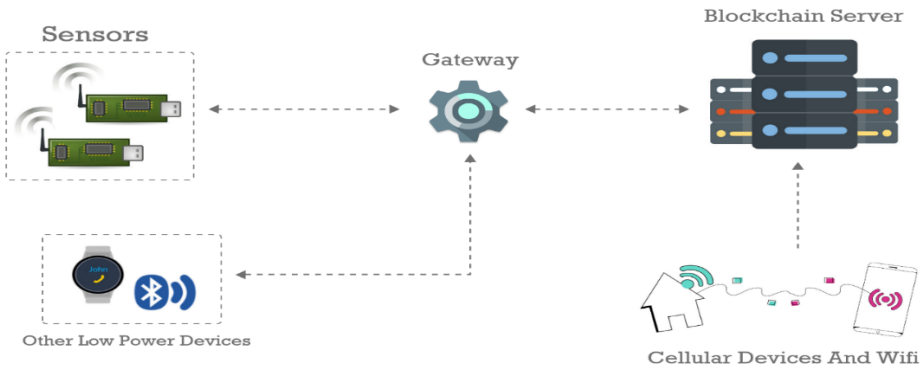

Figure 2.7. Blockchain in IoT

ii. Diversified Services Consensus Protocol

The consensus protocol discussed is the merge of Proof of stake and proof of work. Ethereum Casper FFG uses the same protocol. The implementation of the protocol begins with the genesis block. An appointing committee should be organized which will pay the deposit prior. However, the amount to be deposited depends on the actual situation. Instead of each common block, this paper focuses on the checkpoints blocks. Members vote for the final winner of the fork. The period in which voting takes place is called an epoch. The appointing committee members can also issue transactions just like other nodes of the network. Additionally, they have the charge of voting at justified checkpoints. Ballots are used by the members in voting where they decide which checkpoint block should be included in the main graph. Then the result of the vote is broadcast to the whole network. If BCP1 (Block checkpoint 1) gets more than 2/3 ballots, the block is acknowledged and prepared and the epoch ends. The transactions that are in leaf blocks and those incompatible transactions are sent to the pool for further processing. In the next cycle, BCP1 is committed and finally confirmed once BCP2 is voted and said to be prepared.

f) Consensus in Big Data

Bitcoin is the most known application of blockchain and yearly its size has increased by 15 GB. Since blockchain is a transaction database distributed among multiple nodes which sort of pushes it to the big data territory. **Figure 2.8.** shows a general picture of big data in the blockchain. Blockchain itself has been used in big data architecture using its consensus protocols.

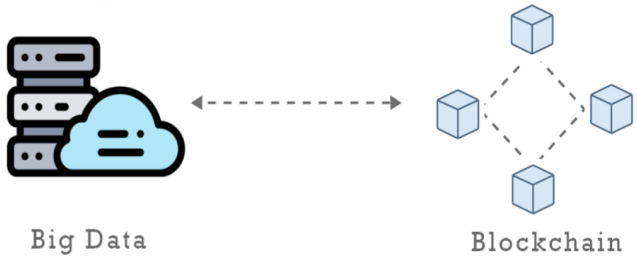

Figure 2.8. Blockchain in Big data

i. Paxos Algorithm for Consensus

Paxos is a decentralized consensus algorithm that lets nodes communicate through an asynchronous network. This algorithm is designed in a way that any value accepted by the majority will not be changed so that all the nodes have the same copy of the value [75].

ii. Proof of Collaboration

The Proof of Collaboration is used in "Making Big Data Open in Edges: A Resource-Efficient Blockchain-Based Approach" here if an edge device (node) intends to generate a new block it will have to show the collaboration from other edges rather than solving puzzles. This comparatively consumes and requires less computation power than puzzle-solving.

g) Consensus in Cellular Networks

Blockchain offers great opportunities for different platforms and applications. Mobile networks can also use blockchain as a part of the infrastructure. Which will provide verifiable and secure digital transactions and can also improve privacy.

i. Proof-of-Location

In the cellular network blockchain, mobile phones are considered nodes. The geographical location of a node at a certain time is attested by a digital certificate, which is known as proof of location (PoL). The Proof of location consensus is required to achieve the proof of location certificate. In Proof of location consensus, there is a "Prover" node and a "Witness" node. The prover node collects proof of location from its close neighbor devices through short-range communication technologies. The witness is those nodes that provide proof of location to the prover [76]. **Figure 2.9.** describes the scenario.

h) Consensus in Social Networks

Social networks are rapidly producing personal data. According to a recent report Facebook which is the largest social network has collected personal data of 300 petabytes since its foundation. Blockchain is a viable solution to protect data by providing decentralized privacy.

i. Proof of Credibility

The contractor signs contracts with different parties and the number of these parties are known as measuring credibility score. Miner in this consensus of blockchain provides proof that he has a high credit score. This proof is an improvement of the previous work of researchers, in which the trust score was on how many good actions a node has made. The improvement calculates the connection between nodes by credibility score. Instead of using proof of stake, the credit score gives a problem

that even if a contract is true or fake the credibility score is added. An attacker can

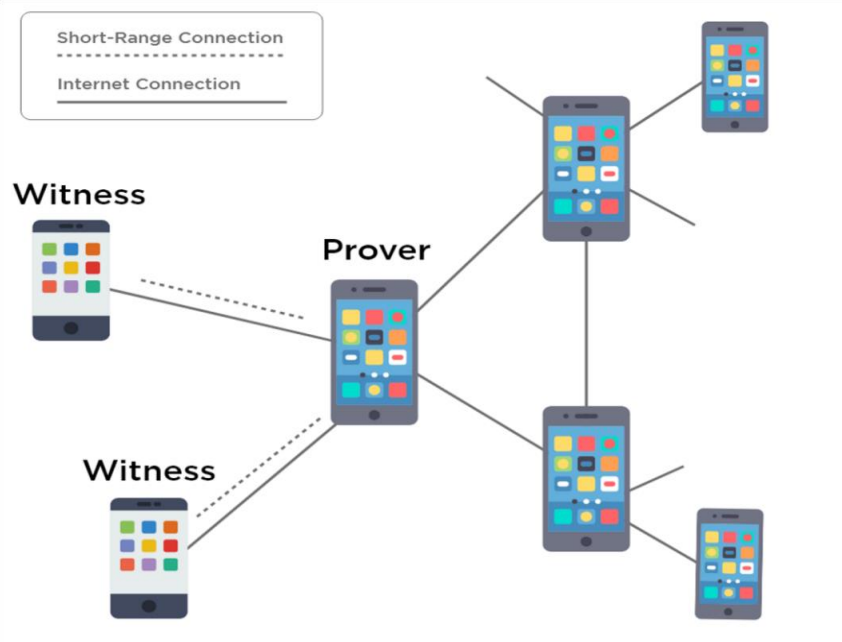

Figure 2.9. Proof of Location

succeed a 51% attack if he makes fake contracts with false parties to increase his credit score. Later this attacker can join the true parties who will renew the contracts illegally. To settle this issue a hybrid of proof of credibility and proof of stake is used in this proposed methodology where these proofs are executed alternately. If a miner generates a block using proof of credibility, the next miner will generate a block using proof of stake [77].

i) Consensus in distributed web services and storage

If cloud storage and web services are made permissionless, decentralized, and secured, it can bring a positive change in the standards of unit economics of decentralized storage. Human capital costs as well as high mark-ups will be eliminated. Solana aims

to bring this trend to the future of blockchain by introducing a scalable consensus called Proof of History

i. Proof of History

Each transaction in a network is time stamped. This protocol verifies the order and duration of time between events. Leaders are designated by the system that can organize and send the user messages to other nodes for further processing. Verifiers execute the transactions and publish the confirmed transactions using computed signatures. These confirmed transactions are votes for the consensus algorithm[1].

j) Consensus in Governance

Blockchain can be used in governance though it cannot take full control from the central body, it can be part of it. Being a part of the governance blockchain can be used in these areas such as voting, transparent budgeting, replacing paper-based systems, and secure data entry as shown in **Figure 2.10**. There are other areas where blockchain can be used, such as digitizing the currency by using cryptocurrency instead which might never be accepted and implemented. There are a lot of consensus protocols such as proof of work and proof of stake that can be used in blockchain in governance.

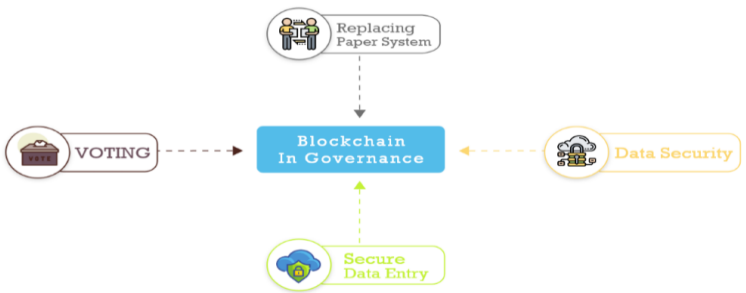

Figure 2.10. Blockchain in Governance

[1][Online]. Available: https://iconetwork.io/tag/proof-of-history

k) Consensus in Entertainment

Blockchain is also very useful in entertainment business such as "Music On The Blockchain" [78] which protects the copyright information of music by storing it on a blockchain. The proof of work and proof of stake consensus algorithms are discussed for the implementation of Music on the Blockchain. More applications of blockchain in entertainment are shown in **Figure 2.11.**

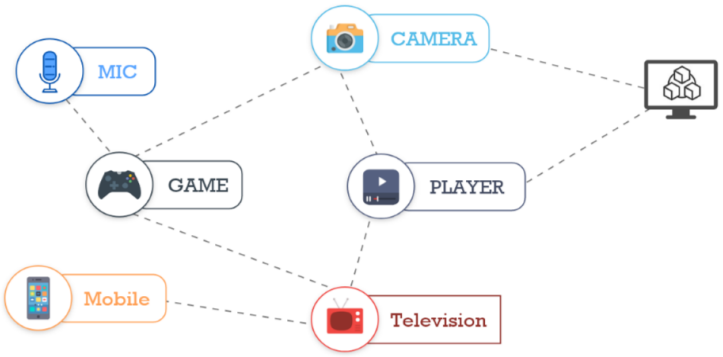

Figure 2.11. Blockchain in Entertainment

l) Consensus in Real Estate

Blockchain can be implemented in real estate such as recording of properties titles and deeds [79]. To ensure the safety of the data it is designed to be stored in the blockchain and to ensure safe transactions on that blockchain proof of work consensus has been favored. Also digitizing the land record system by blockchain would make it secure from corruption [80]. The proof of existence is advised to be used for intellectual properties. Other than that proof of concept is also used in blockchain for real estate [81]. The general image for real estate over the blockchain is shown in **Figure 2.12**.

Figure 2.12. Blockchain in Reak Estate

m)Consensus in Power

Blockchain is going to change the legacy systems of centralized nature by hybrid distributed systems which are made up of solar power microgrids and large power plants. This distributed energy system will be a reliable, efficient, and renewable energy delivery system. There is also a possibility that blockchain may change the trading system and businesses would trade-off using electricity for example if a factory needs additional power, it can buy unused electricity which another factory is selling for five minutes. These five minutes are the unused downtime minutes of a factory. This trade can give efficiency benefits to grid operators.

Each home is installed with Smart Meters (SM) to achieve better scheduling in the smart grid. These SM collect real-time data of electricity consumption and utilities use this data to provide smart home services in a better way. The real-time data can disclose the private data of the user and an adversary can take advantage of this data by reading the usage patterns of the electricity consumption profile of the user. In [82] a privacy-preserving and data efficiency aggregation scheme is proposed via blockchain.

n) General Consensus Algorithms

The algorithms discussed in this section are general purpose which can be used for any type of asset. **Table 2.2.** has some of the important features of the consensus algorithms.

Table 2.2. Features of General Consensus Algorithms

S.No	Consensus Algorithms	Permissioned / Permissionless	Platform	Applications	Programming Language	Advantages	Disadvantages
1	Proof of Vote [83]	Permissioned	Not Known	Consortium Blockchains	Any programming language can be used	Consistency, Availability, Partition tolerance	The problem in modular design and parallel processing
2	Tendermint [83]	Permissioned	Cosmos, Ethereum	ABCI(Application Blockchain Interface)	Rust, Go, Haskell, C/C++, Java etc	Does not require mining	Unfair
3	Proof of Human Work [81]	Not known	Humancoin	Cryptocurrency, password protection, Bot detection)	Not known	Fair	Requires an initial trusted setup, Prone to malicious attack
4	Simplified Byzantine Fault Tolerance [83]	Permissioned	Chain	Financial Applications	Java, Node, Ruby	Good security, Signature validation	Not for public blockchain
5	Practical Byzantine Fault Tolerance [85, 86]	Permissioned	Hyperledger, Zilliqa	Cryptocurrency and other asynchronous systems	Golang, Java	High transaction throughput	Centralized
6	Raft [87, 88]	More suitable for Private/ Permissioned Blockchains [62]	Kaleido, IPFS Private Cluster, Quorum	CockroachDB	Go, C++, Java, Scala, and Rust	Supports configuration changes, Simple and easy as compared to Paxos	Centralized
7	Proof of Property [89]	Not known	Ethereum	Not known	Not known	Scalable, Save a lot of local space	Forking may create issues

i. Proof of Vote

A Proof of Vote is built for consortium blockchain where companies develop a partnership and each company represents an officer [84]. These companies share their business-related data via a coalition committee. Business transactions and operations

are recorded on the blockchain. The companies do not agree on giving the right to any of the companies to produce a block so they decided to hire a butler team. The team is hired from all over the world and the election is held among them regularly. The blocks are produced by the team and are sent to the companies for verification and voting which makes the power decentralized among the team. To maintain reliability, safety, and efficiency the butlers are paid high salaries by the companies. Anyone can join the butler team but the team member is recommended by the coalition and they have to submit a deposit. The members of the coalition supervise the work of a butler and grade accordingly so that only the honest one can survive. So PoV is a consensus method proposed for consortium blockchain which is maintained by organizations and enterprises in different areas of the world.

ii. Tendermint

Tendermint is a permissioned consensus algorithm. It is the same as PBFT as it can tolerate one-third of malicious nodes [85]. In tendermint protocol, the participants are known as validators who vote on their proposed blocks. There are two steps of voting pre-vote and pre-commit. When more than two-thirds of validators pre-commit in the same round for the same block, it is then added to the blockchain.

iii. Proof of Human Work

Puzzles related to Human work are very much similar to PoW except that a human is involved in finding a solution. The problem solver should not be a machine only and this is the main difference between PoW and PoH. A puzzle should be made hard to solve for humans and uneasy to solve for machines. Although we expect the verification to be easy for machines as in PoW [83].

iv. Simplified Byzantine Fault Tolerance

In SBFT block gathers all the transactions, batch them and validate them in a new block [86]. All the nodes follow the rules of a block generator to validate all the transactions. A block signer validates these transactions and adds its signature. So if any of the blocks misses one of the keys, it is rejected.

This algorithm uses an adapted version of the Practical Byzantine Fault Tolerant (PBFT) consensus algorithm. This protocol is also aimed to provide improvements over PoW. There is a single validator who is a known party and the nature of the ledger is permissioned. The validator forms a new block with a bundle of proposed transactions. Consensus is achieved when a minimum number of nodes approve a block. The number of nodes to reach consensus is 2f+1 that has a 3f+1 number of nodes where f is the number of faulty nodes. For example, if a system has 7 nodes and 2 of them are faulty then 5 nodes must agree.

v. Practical Byzantine Fault Tolerance

A solution to the Byzantine general problem is the Federated Byzantine Agreement. In this approach, every node knows each other and knows which one is important and which is not [87]. PBFT is an algorithm that uses this approach. Hyperledger uses this principle and its consensus algorithm. The approach is like multicasting. A primary is responsible for sending requests to other nodes in its group. The service is said to be approved if 1/3 different replicas (nodes in the group) approve the receipt of the requests. If the client does not receive any replicas it will send requests to all the nodes instead of sending them to the primary only. This would be the case if the primary is faulty [88].

vi. Raft

Raft can accommodate 50% of malicious nodes [89]. It is a voting-based consensus algorithm that is composed of two stages: leader election and log replication [90]. The

ordering of transactions is a task given to the leader. When an existing leader fails, a randomized timeout for each server selects another leader. This way a leader is chosen. After a leader is chosen a replication stage starts. In this stage, the leader makes its version of the transaction log by accepting log entries from clients and broadcasting these transactions. This method has low latency and high throughput. The performance and throughput are dependent on the leader node so if the leader node is infected the whole system will be destroyed. It is not appropriate for IoT because of its low security and restricted throughput.

vii. Proof of Property

This proof allows participants not to have a local copy of the full blockchain [91]. The owner of the new transaction should have coins enough in their address to fulfill the transaction. New participants can validate the transactions without the need to download a blockchain initially. The new participants need to access the root hash of the Patricia tree system state of the new block. As members of the network can get a state from the header, they can delete the old body of the blocks and save a lot of local space which isn't the case in traditional blockchain applications.

2.1.5. Research Issues and Challenges

In the following subsection, we discuss blockchain challenges and issues.

a) Denial of Service attacks

The attacker crash a node by flooding a large amount of traffic in DOS [92]. It prevents authorized users from retrieving the service or resources. Similarly, Distributed Denial of Service (DDOS) is another type of attack where a node is flooded with malevolent requests [93]. In DDOS multiple attackers attack a single node.

b) Sybil attacks

Using multiple identities to attack a larger portion of the network is called a Sybil attack. The invaders can launch numerous false nodes that seem honest to their peers. These false nodes take part in falsifying the network to authenticate illegal transactions and to modify valid transactions. They can use virtual machines, several devices, or Internet Protocol (IP) addresses as bogus nodes for the attack. The P2P network assumes that every participating node contains only one identity. Thus, numerous forged nodes give attackers the ability to repudiate transmitted blocks and to outvote authentic nodes. When an attacker controls a large number of nodes in the network, it increases the chances of double-spending [94, 95].

c) Eclipse Attacks

In an eclipse attack [96] specific nodes are isolated from the peer-to-peer network by the attacker. Similar to Sybil attacks, it does not attack the entire network. Once the target node is isolated, the attacker controls all outgoing connections of the node [97]. From there on, the attacker can abuse the target network and dispatch distinctive sorts of attacks on blockchain mining power and agreement components. These attacks include double-spending, engineering block races, selfish mining, and splitting mining power.

d) Routing Attacks

In a routing attack [98], a message is intercepted by the attacker in the blockchain network. The attack alters the message and sends it to its neighbors. Furthermore, this attack is divided into partitioning attacks and delay attacks. In a partitioning attack [99], the entire blockchain network is divided into two or more portions. Whereas, in a delayed attack, the attacker captures the message and tampers with it. Then, it redirects

the tampered message to another BC network portion. Different consensus mechanisms are used by blockchain to develop trust among blockchain peers. However, there are some possible attacks on these consensus mechanisms.

e) 51% Attacks

A miner, having 51% or more hashing power, can initiate a 51% attack in the blockchain network. The 51% attack enables the attacker to stop the confirmation of a new block [99]. Additionally, the attacker can reverse transactions already confirmed by the blockchain.

f) Double spending

In double-spending [100] multiple transactions with the same cryptocurrency are performed by a user. This transaction is broadcast to each node in that network. This transaction needs to be confirmed by the nodes, this confirmation is time consumable [101]. This running time between two transactions' initiation and confirmation can be a window for the attacker to quickly launch his/her attack.

g) Alternative History Attacks

In an alternative history attack, a transaction is sent to the merchant by the attacker [102]. In addition, a double-spending transaction is included by the attacker in an alternative blockchain fork [103]. The merchant sends the product after n blocks confirmation. So, the attacker tries to find more than n blocks. If the attacker succeeds, he gains his coins by releasing the fork.

h) Race Attacks

In the race attack, the attacker creates two transactions. The first transaction is sent to the merchant by the attacker [104], [105]. This product is sent by the merchant without

confirmation. Meanwhile, the second transaction is broadcasted by the attacker to invalidate the first transaction.

i) Finney attack

In the Finney attack [106] two similar transactions i.e., one crediting the target and the other crediting the attacker are used by the attacker. This attack mines a block that includes the first transaction and delays publishing it. Meanwhile, the attacker mines the second transaction. When the attacker succeeds, he purchases goods with the first transaction. Then, he releases the pre-mined block which includes the first transaction. The attacker receives both goods and coins whereas the merchant finds their transaction invalid [107].

j) Blockchain-Based Research Issues in Healthcare

Blockchain has improved the medical and smartphone applications but there are still some security issues as blockchain comes with its potential problems. Any industry including healthcare that needs to use blockchain and its devices should train themselves in these areas to improve it. Such education can improve patient-centered data [108]. The blockchain experiments in this research need a patient to authorize himself before transferring a record and this could lead to threats. Key leakage and management is another issue that is not addressed. If a key is lost by a patient the data is difficult or rather impossible to be authenticated or recovered. A mechanism should be discovered for recovering data.

k) Blockchain Research Issues/Future Work in Real Estate

Using resource information by a customer can lead to conclude the business work of other customers. So a hybrid blockchain that includes both private and public best

features can be deployed to audit the access of data as it provides the best authority system for the participating nodes [109].

2.2. Intelligent Transport System

Traditional transport systems face several issues that including air pollution, traffic congestion, and high accident rates. Due to such problems, researchers came up with the idea of integrating virtual technology with the traditional structure of transportation, hence naming it as Intelligent Transport System (ITS). ITS is an evolving technology that has several applications like traffic management and congestion control for example [110] if an accident has occurred somewhere on a road, vehicles can alert each other to avoid traffic jams. Privacy and Security is the vital right of each resident and must be incorporated in every system. Privacy ensures that the real identity of an individual/vehicle/node is not observed while security assures confidentiality, authentication, availability, and integrity of the messages. Vehicles share basic safety information regarding location, speed, and other personal information with each other. An ITS comprises of Intelligent Transport System-Stations (ITS-Ss), which can be vehicles, Road Side Units (RSUs), and servers. Vehicles are equipped with On-Board Units (OBUs) through which they can communicate with other stations.

To achieve communication securely and privately, the safety properties must be safeguarded i.e. integrity, authentication, privacy, and non-repudiation. Vehicle status data, for example, velocity, speed, and direction are traded occasionally through Cooperative Awareness Messages (CAM) and are utilized to improve safety and traffic productivity. Eavesdroppers may get the benefit of user tracking by having access to user status information [111].

2.2.1. Privacy in ITS

In ITS digital certificates are issued to each vehicle for communication by Certificate Authority (CA) which is a third party [112] according to IEEE 1609.2. The revocation process is done in case a vehicle is involved in malicious behavior and their certificates are revoked. Vehicles need to be authenticated before they take part in the network in ITS. Privacy protection of the vehicle/user is also necessary as the malicious vehicle may take advantage of the genuine identity of the vehicle/user. Pseudonyms can be used to protect the vehicle from malicious attacks [113]. It is suggested that pseudonyms should be changed on regular basis to avoid traceability [114]. In ITS identity and location privacy, both are required to escape wrong use by malevolent ITS-Ss. Pseudonym-centered techniques discussed in [115] use cryptography for user identity protection. These techniques however generate high communication and computation overhead.

2.2.2. Blockchain-Based Research Issues in ITS

This section discusses different blockchain-based techniques and schemes used by researchers for achieving privacy and security. **Table 2.3**. summarizes these techniques. The work suggested by Wang et al. [116] builds a reputation service where deceivers who generate false messages are identified. There should be two servers one for reputations and the other for pseudonyms. The server checks and compute vehicles reputation suffers extra overhead and causes delays in communication.

Table 2.3. Blockchain schemes used in Intelligent Transportation Systems

Reference no	Authors/ year	Method	Summary of Method
[116]	J.Weng	RPRep: A robust and privacy-preserving	Builds a reputation service via blockchain where deceivers who

	et.al. 2016	reputation management scheme for pseudonym-enabled VANETs	generate false messages are identified in ITS network.
[117]	U.Rajput et al. 2016	Hierarchical privacy-preserving pseudonymous authentication protocol for VANET	The idea of a primary and secondary pseudonym is given using blockchain where Primary pseudonyms are managed by CA. Similarly, secondary pseudonyms are used for V2X messages communication and generated through RSUs.
[119]	H.Li et-al./2019	Blockchain meets VANET: An architecture for identity and location privacy protection in VANET	The blockchain-based VANET resolves the problem of centralization and develops trust between entities in the existing VANET
[120]	M.Wagner 2018	Cyber-physical transactions: A method for securing VANETs with blockchains.	A blockchain architecture proposed is used for safeguarding VANETs without dependence on RSUs.

[122]	Z.Lu et al. 2018	Bars: a blockchain-based anonymous reputation system for trust management in VANETs	Blockchain-based Anonymous Reputation System (BARS) is suggested for preventing the distribution of counterfeit messages from authentic vehicles
[123]	Ao Lei et.al	A blockchain-based certificate revocation scheme for vehicular communication systems	A certificate revocation scheme for Vehicular Communication Systems (VCS) via blockchain is proposed which prevents insider attacks. This scheme reduces the overhead of broadcast messages and Certificate Revocation List (CRL) size.
[124]	A. Li et al. 2020	A traceable blockchain-based access authentication system with privacy preservation in VANETs,	BC for VANETs is, introduced in cloud servers. No information linkable to real identity is included in transactions.
[125]	L. Benarous et al. 2020	Blockchain-Based Privacy-Aware Pseudonym Management Framework	The proposed system is a scheme comprising of two Blockchains. In permissionless blockchain certified pseudonyms are saved, and

		for Vehicular Networks	the revoked pseudonyms are spared in a public and permissioned blockchain.
[126]	K.Si et al. 2020	Blockchain-based multimedia sharing in vehicular social networks with privacy protection	A blockchain-based privacy-preserving scheme is proposed for data sharing of multimedia in VSNs.
[127]	J. Cui et al. 2021	Secure and Efficient Data Sharing Among Vehicles Based on Consortium Blockchain	A consortium blockchain technology is deployed to achieve anonymous and traceable vehicle-to-vehicle (V2V) data sharing, to prevent second-hand sharing of data effectively.
[128]	J.Ma et al. 2021	Attribute-Based Secure Announcement Sharing among Vehicles Using Blockchain	A blockchain scheme using RSU for an attribute-based encryption algorithm is proposed. In their scheme, the target is to protect the open cloud access environment from unauthorized access.
[129]	S.Bao et al. 2019	Pseudonym Management Through Blockchain: Cost-	There are mainly two contributions in the paper, first is the use of blockchain technology for

		Efficient Privacy Preservation on Intelligent Transportation Systems	pseudonym management and the second is the VCS pseudonym certificate shuffle scheme. It reduces the cost of pseudonym management and generation.

The idea of pseudonyms as primary and secondary is discussed in [117]. The primary pseudonym is a single point of attack as it is provided by the Certificate Authority (CA). Similarly, secondary pseudonyms are used for V2X messages communication and generated through RSUs. RSUs however are prone to side-channel attacks as they are located in the open infrastructure [118]. The blockchain-based VANET resolves the problem of centralization and develops trust between entities in the existing VANET. However, the efficiency of the privacy protection mechanism is always an open issue in VANET [119].

An architecture proposed [120] is used for safeguarding VANETs without dependence on RSUs. However, the protocols presented have scalability issues as it uses Proof-of-Work and it has computational overhead. Consensus algorithms like proof of work and proof of stake are facing some issues. For example, proof of work consumes too much energy and electricity while proof of stake deals with the phenomenon that the rich become richer [11]. The data privacy techniques mostly depend on complex cryptographic algorithms; therefore, they are inefficient and hard to measure with large applications [121]. Research has been going on reducing the complexity and enhancing the efficiency of these cryptographic techniques.

The issues related to the trust and privacy of VANETs are addressed in [122]. A Blockchain-based Anonymous Reputation System (BARS) is suggested for preventing the distribution of counterfeit messages from authentic vehicles. This system keeps the identity of the vehicle protected. The public key of the vehicle acts as a pseudonym to provide anonymous communication and to prevent linkability between a public key and real identity. An algorithm for reputation management is designed to avoid the distribution of forged messages while the vehicles get an incentive to reveal the misbehavior of other vehicles.

The work of Ao Lei [123] proposed a certificate revocation scheme for Vehicular Communication Systems (VCS) via blockchain which prevents insider attacks. This scheme reduces the overhead of broadcast messages and Certificate Revocation List (CRL) size. In this Blockchain structure, the PKI distributes the CRL and keeps track of the ownership of pseudonym sets efficiently. Limitations of this research are: further investigation needs to be done for the efficient blockchain consensus algorithms and further evaluation of the private security and accountability.

Privacy and authentication of users [124] are two important VANET problems. It is important to avoid internal vehicles from spreading forged messages while at the same time preserving vehicle's privacy against tracking attacks. The third important point is to remove dependency on third parties. BC for VANETs is, introduced in cloud servers. No information linkable to real identity is included in transactions.

PKI as a central authority is prone to attacks. The proposed system is a scheme comprising of two Blockchains [125]. In permissionless blockchain certified pseudonyms are saved, and the revoked pseudonyms are spared in a public and permissioned blockchain. The first one is accessed by the RSUs with read rights and overseen by the enrolled vehicles. The second blockchain is supervised by the vehicles

with read-only rights and the RSUs (having write access rights). RSU is responsible for managing and accessing the offline chain which is private. VPKI in the existing literature is maintained by the RSU, while the solution in this paper changes the load between the Vehicles and RSUs. The goal is to decrease the dependency on trusted third parties in terms of maintaining and generating pseudonyms. The task is performed by managing pseudonyms over RSUs and vehicles computational and storage resources by using blockchain technology. The shortcoming of this paper is the complexity of implementation.

The problem defined in [126] is both the vehicles' and users' privacy may be disclosed, and the communication routines of users are easily evaluated by adversaries. The shared multimedia data is more prone to being tampered with or altered. A blockchain-based privacy-preserving scheme is proposed for data sharing of multimedia in VSNs. The use of cryptography measures is used to cover the real identity of RSUs and vehicles. Blockchain is also used to protect data from tampering and forging by malicious users and safeguard consistent data sources. Users use a pseudonym (ID) to communicate during multimedia data sharing. The experiments and security studies prove that the proposed scheme achieves data integrity, verification, privacy protection, and efficiency. Shortcomings of this research are that the damage attackers can cause to the RSU should be paid more attention and work should be done for the security and performance of RSU. RSU can be curious to know the actual identity of the vehicle and to know the communication routines of the vehicles. Advanced privacy protection should be achieved i.e. location privacy. A consortium blockchain technology is deployed by J. Cui et al. [127] to achieve anonymous and traceable vehicle-to-vehicle (V2V) data sharing, to prevent second-hand sharing of data effectively. The combination of Blockchain and 5G is used to

share data without relying on RSUs. The consensus used is delegated proof-of-stake algorithm which is prone to attacks discussed above. The scheme proposed by J.Ma et. al. [128] is a blockchain scheme using RSU for an attribute-based encryption algorithm. In their scheme, the target is to protect open cloud access environment from unauthorized access."

Pseudonym management is done by PKI which is a central authority and the central system is always prone to attacks, has low scalability, and is highly unstable. Different locations require a different number of pseudonyms so this becomes an issue of how to distribute the pseudonyms according to traffic. Pseudonym shuffle done by RSU is suggested but it creates a burden on RSU. Privacy-preserving pseudonym management framework S.Bao [129] is more cost-effective than the existing approaches. There are mainly two contributions in the paper, first is the use of blockchain technology for pseudonym management and the second is the VCS pseudonym certificate shuffle scheme. It reduces the cost of pseudonym management and generation. Privacy manager (PM) which is distributed is also introduced which aims to improve the network robustness and ease the computation burden on RSUs.

If a malicious user compromises a PM or if a PM loses its link with a blockchain, the entire blockchain will remove it. The PKI will abandon all the pseudonym sets and will not use them again. The ETSI standard uses 288 × 100 = 28, 800 pseudonyms each day which makes 864, 000 per month. The proposed system uses 100, 000 pseudonyms with the storage capacity of 1,000 pseudonyms certificates. Total processing time varies from 0.2 seconds for 100 transactions to 2 seconds for 1,000 transactions. The problems identified are: The shuffle management in a cloud is not explained clearly i.e which protocol is followed by PMs for shuffling pseudonyms. The solution for pseudonym shuffling in situations where there is more

traffic than expected (such as in the case of VIP with high protocol) is not provided. Proof of Work is used in this approach which is time taking protocol.

Consensus used in blockchain-based pseudonym shuffling is PoW which takes heavy computational power of the resources. Another consensus is suggested for pseudonym management over blockchain but they need to be tested and compared. Secondly, curious RSU will try to analyze the communication habits between different real vehicles and the user's real identity behind the vehicle. External attackers can attack RSU by stealing the data stored in RSU. An attacker may forge a large amount of communication data within RSU and may perform other illegal operations. Hence RSUs are at risk. The next chapter provides a detailed discussion on how to overcome the issues related to pseudonym management.

2.3. Summary

This chapter contains a detailed discussion on blockchain, its features, types, and applications. The blockchain consensus methods are discussed in detail dividing them into cryptocurrency consensus and general where the reader can take advantage of which consensus is suitable for their application. Blockchain-related attacks and challenges are also part of this chapter whereas the research issues in ITS and blockchain are identified.

Chapter 3: Proposed Architecture

3.1. Proposed Work

We propose a pseudonym shuffling scheme that aims to reduce the cost of generating and pseudonym management for ITS via an efficient consensus mechanism. The architecture follows the traditional structure of Vehicular Communication System (VCS), where a central manager is Public Key Infrastructure (PKI) or a Certificate Authority (CA) is designed to maintain pseudonyms certificate centrally. We further elaborate the components of the scheme as follows.

a) **Public Key Infrastructure:** Public Key Infrastructure (PKI) is at the top of the hierarchy. PKI is a central trusted authority used to provide cryptographic pairs of keys, identity for a long time to all the authentic vehicles and infrastructures and certificates. PKI manages several Privacy Managers that are dispersed in a sparse manner in a network. PKI in our scheme is used to allocate pseudonym identities and certificates revocation.

b) **Privacy Managers** Privacy Managers (PMs) are devices introduced that come after PKI in the hierarchy. The purpose of Privacy Manager is to improve robustness in the system and act as miners of the blockchain.

c) **Communication Paths:** PM is a multi-access edge computing node in a network. The data gateway is placed between this computing node and the radio access network that forwards data from the multi-access edge computing node to the radio access network and user equipment. The radio access networks use for example base stations with 5G technology. The user equipment could be mobile phones or vehicles with On-Board Units (OBUs). User equipment accesses the multi-access edge node through an air interface between the radio access network and user

equipment.

Transactions in the blockchain contain asymmetric cryptographic material that is transaction is signed by the source using its private key and encrypted using the receiver public key. This makes the pseudonym shuffle path more secure, as a result, attackers or other unintended PMs cannot get any useful information.

d) **PM Cloud:** Cloud is used by PMs to manage blockchain among Privacy Managers.

e) **Road Side Units:** Road Side Units (RSUs) are fixed infrastructures placed geographically in an ITS network to communicate with vehicles and other RSUs. In our scheme, RSUs will distribute the pseudonyms according to the distributed algorithm and are placed third in the hierarchy as discussed in [129].

f) **Blockchain:** Blockchain in our scheme is used to record the pseudonym shuffling results in the form of transactions. The transactions contain the timestamp and the key pairs of the sender and receiver PMs along with the data encrypted with the receiver public key and signed by the sender/source private key.

PKI is accessed only twice in the system, first, it provides the initial permanent identities and certificates to vehicles, and the second time it is accessed in case of malicious activity for certificate revocation [131]. The RSUs and PMs have devices that can communicate wirelessly over the wireless medium, using VCS communication standards (C-ITS or DSRC). RSUs act more as the communication bridge between vehicles (users) and service providers. Furthermore, vehicles are equipped with On-Board Unit (OBU) which are computerized devices for provisioning VCS standards. A PKI comprises Certificate Authority (CA) and other third-party services.

PKI is responsible for all the cryptographic credentials for pseudonyms such as pseudonym certificates, key pairs, and anonymous identities. PM manages certain areas under its security domain. PMs are supposed to support the PKI to manage the

security domains and their cryptographic material that are placed beneath the PKI layer. This scheme proposes to install PMs geographically sparse. There is continuous communication between vehicles V2V (Vehicles to Vehicles) and V2I (Vehicles to Infrastructure). The infrastructure is RSU which collects safety-related messages from vehicles at regular intervals. The safety message contains the current status of the vehicle which includes the position, orientation, speed, and direction along with pseudonyms and time stamps. Vehicles contain a set of pseudonyms to use in diverse time durations in VANETs. Pseudonyms need to be used for a short period and shift to new ones often instead of achieving privacy. The vehicles do not exchange pseudonyms with each other but they get their updated sets from the sets given to them by RSUs.

The significance of the proposed work is providing an optimal solution for pseudonym management via an efficient consensus mechanism for pseudonym shuffling. External attackers and curious RSUs cannot take an advantage of data kept in BC as the transactions at RSU are anonymous. The scheme provides pseudonym shuffling at zones where there is maximum traffic. For example traffic signal stops, roundabouts and car parks, etc. The pseudonyms sets are given to vehicles and a percentage threshold is set over the used pseudonyms such that if a vehicle hands over the used pseudonyms, it is left with enough pseudonyms to use in the meantime. For example, a vehicle contains 100 pseudonyms and it uses 90 out of them, it will hand over the 90 used pseudonyms to RSU and is left with 10 pseudonyms to use while waiting for the shuffle round. These pseudonyms are encapsulated in a package and encrypted with the public key of the destination and signed by the senders. The vehicles also get enough new pseudonyms to reduce the overhead of transmission which is created by allocating the pseudonyms frequently for a short period. We know that a blockchain is a decentralized distributed network that provides security, immutability, transparency,

and privacy. There is no concept of centralization to verify and validate the transactions, but still, transactions in the blockchain are considered to be completely verified and secured. This is the result of a core algorithm present in every blockchain network called consensus protocol. A consensus algorithm is a technique through which all the peers of the blockchain network reach a common agreement about the current state of the distributed ledger [1]. So consensus algorithms provide trust and reliability among unknown peers in a distributed environment. The consensus mechanism ensures that every new block added to the blockchain is the only truth that is agreed upon by all the blockchain nodes [46].

The blockchain consensus protocol comprises of some specific aims that are coming to an agreement, co-operation, collaboration, mandatory participation of each node in the consensus process, and equal rights to every node. Hence, a consensus algorithm targets finding a common agreement that is a win for the whole network. We have simulated the consensus algorithms which are relevant for privacy in ITS.

3.1.1. Consensus Algorithms

Proof of work is the first consensus protocol used by a public blockchain. All the nodes needed to solve a cryptographic puzzle by brute force. The node which wins the puzzle is rewarded and all the other nodes computations are wasted. The consensus is achieved as 51% of power [32]. PoW provides security to the network in terms of transactions because it takes massive electricity and computing power. Therefore, PoW is observed as preventive from the attacks of Denial-of-Service (DoS) and Distributed Denial-of-Service (DDoS) in the blockchain. Where the PoW provides prevention from the mentioned attacks, it also suffers some issues:

- **Time-Consuming:** It is a time-consuming protocol as it involves solving a puzzle and guessing it takes time. Transaction confirmation can take up to 10

to 60 minutes so the transactions are not instant as it takes time to mine the transactions and commit to the blockchain.

- **Energy consumption:** PoW consensus requires a network of powerful computers to protect the network. These powerful computers are costly in terms of computational power and energy consumption. Miners need this specialized hardware with heavy computing capability to perform mining and be rewarded as incentives. A huge amount of electricity is needed to execute these mining nodes constantly.
- **Vulnerability:** PoW consensus is susceptible to 51% attacks, which means, unfair miners could achieve a more than 50% of hashing power and take advantage of the blockchain by manipulating it.
- **Centralization:** Mining requires specialized and expensive hardware such as ASIC machines with which winning chances get higher. It becomes unmanageable for some miners to grow the expenses, as a consequence, only a little number of miners wins the mining race. The result of this is a steady rise in the centralization of the computer system, as it turns into a game of riches. Hence for real-time scenarios, it is not a suitable protocol. We implemented PoW with two differently designed codes to find out the difference in execution time by changing the difficulty and puzzles. **Figure 3.1.** shows the basic working of PoW. **Algorithms 1 and 2** depict the code design for PoW 1 and PoW 2 respectively and their respective running time complexity which we implemented. **Algorithm 1** takes two inputs P and St where P is a puzzle that needs to be solved by miners whereas St is the string of characters (0-9a-zA-Zspecial characters) (Algorithm line 1-3). Line 4 begins the procedure wherein lines 5 and 6 we initialize the variables P and St. In line 7 we set a for loop on

the string where as in line 8 we guess the puzzle from the given string, if the puzzle is found in the string return puzzle found else return the St. The algorithm takes a whole puzzle to check against the string and we change the puzzle string manually to see how much time the whole string (puzzle) takes for guessing. In **Algorithm 2** line 2- 5 declares four inputs such as h: Hash, d: Difficulty, n_0: Nonce. We begin the procedure from line 7 and set the difficulty level at line 9 where P: Puzzle takes a hash of data (within the block), timestamp (of the block), and a nonce (random number used only once). We set the difficulty i.e. leading zeros over this hash and then we start guessing and monitoring the time it takes with different difficulty levels. We set the guessing character set for both the algorithms to provide the search set for the puzzle to find the string (puzzle) in the given set in line 9 **Algorithm 2**. Line 11-25 begins the procedure for guessing the puzzle within the given character set for each character in the puzzle. If string character within the puzzle is found in character set ch then return P else return string. The results are shown in chapter 4.

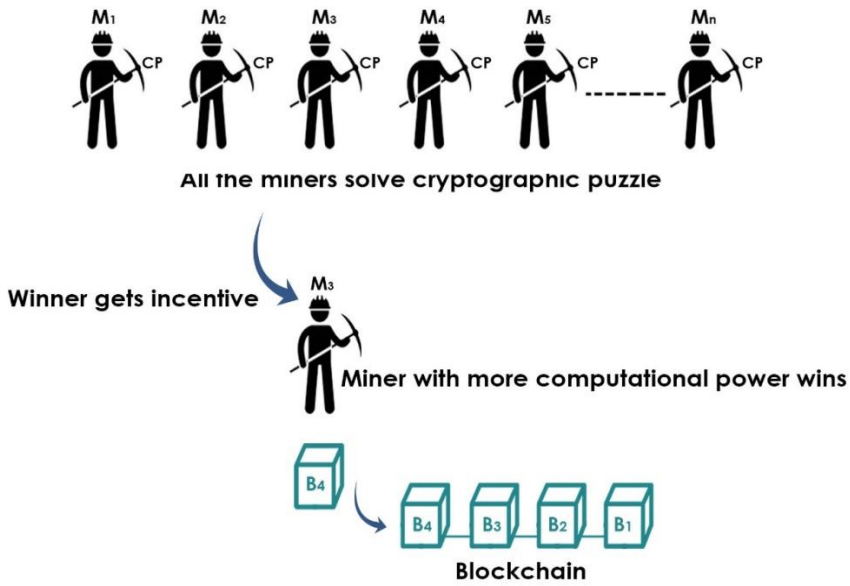

Figure 3.1. Proof of Work

Algorithm 1: Proof_of_Work_1_Consensus Implementation

1. **Inputs:**
2. P: It refers to the puzzle
3. St: It refers to a puzzle string
4. **Begin Procedure**
5. Step-1. St = n (No of Strings)
6. P = 1 (Initialization)
7. Step 2. for (St = 1; St <= n; St ++)
8. Step 3. if (P = St[i])
    i. return (P)

b. else

i. return (St)

9. **End Procedure**

10. **Output:**

11. Return Puzzle Status

Run time Complexity: O(n)

Algorithm 2: Proof_of_Work_2_Consensus Implementation

---

1. **Input:**
2. h: Hash
3. d: Difficulty
4. n_0: Nonce
5. P: Puzzle
6. **Begin Procedure**
7. Step 1:
8. P = h* d * n_0 (difficulty level)
9. Character ch = ABC……Zabc….z0123…9!@#$%&*() {}|
10. Step 2.
11. for (St = 1; St <= i; St++)
12. do
13. {
14. return (P)
15. }
16. while

17. {
18. St[i] = ch;
19. }
20. else
21. return (St)
22. **End Procedure**
23. **Output:**
24. Return Puzzle Status

Run time Complexity: O(n)

3.1.2. Proof of Kernel Work

PoW is an energy consumption consensus that also leads to centralized mining using rewarding structures. So, it appears desirable to preserve the benefits of PoW while also holding its energy consumption and justifying, if not excluding, centralized mining. The Proof of Kernel Work (PoKW) [128] eliminates some of the network nodes for the mining race and randomly chooses nodes (the kernel) for participating in mining. PoKW doesn't stress using particular blockchain technology, hence PoKW is pertinent for a broader range of blockchain applications. However it also suffers some issues: If an adversary knows a private key ski, it can duplicate the key and continue in a parallel private current thread of mining race. This way an adversary can get an advantage of winning a race. PoKW also involves guessing a puzzle and consuming time and resources. **Figure 3.2.** explains the process of Proof of Kernel Work.

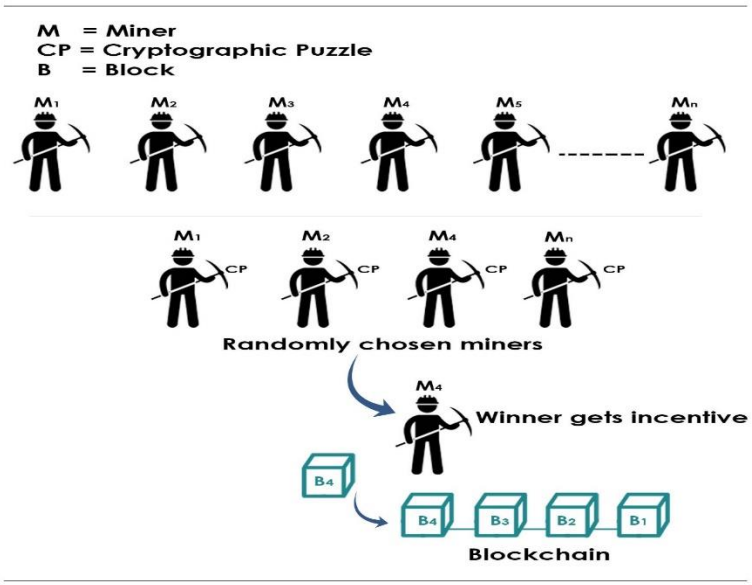

Figure 3.2. Proof of Kernel Work

3.1.3. Proof of Elapsed Time

In PoET each node is given a randomized timer object from a trusted code. The node having the shortest timer is when expired the node wakes up, propagates a signed certificate to show this node is the block leader. The timer is given randomly so that the malicious user does not try to continuously get the shortest timer [58]. PoET is good in terms of winning block time duration but it has some disadvantages: It requires dedicated hardware having intel SGX. It also suffers Sybil attacks as an adversary gets a hold on the nodes with the help of forging multiple nodes identities. PoET is also prone to Denial of Service (DOS) attacks in which the adversary floods the network with requests. The working of PoET is given in **Figure 3.3.** followed by **Algorithm 3** which explains the working flow of PoET which we implemented in node.js and its running time complexity. In this algorithm, the client nodes generate random time ts (lines 8 and 9) without any threshold value and the nodes time is compared with each other (line 11). The node with the smallest time is the winner of the block.

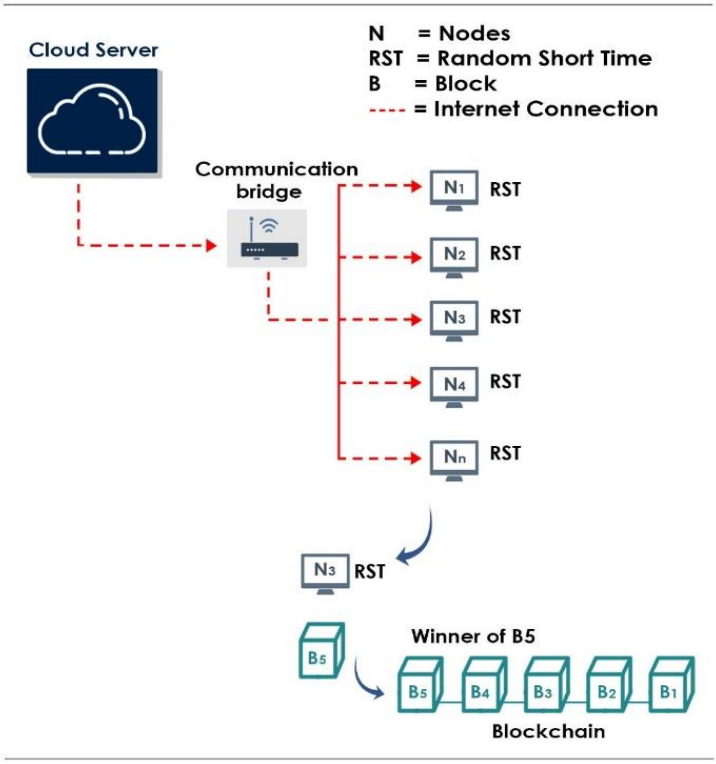

Figure 3.3. Proof of Elapsed Time

Algorithm 3: Proof_of_Elapsed_Time_Consensus

1. **Inputs**:
2. Nn: Total number of client nodes
3. Nc: Client node (Available objects)
4. **Begin Procedure**
5. Step-1.
6. t = time_slot
7. Step-2.
8. ts = Computer node (smallest time)
9. ts = 1 (initialized by 1)

10. Step-3.
11. for (ts <= Nn; ts++)
    a. if (ts < t)
        i. ts = current time
    b. else
        i. ts ! = current time
        ii. continue
    c. return (ts)
12. **End Procedure**
13. **Output:**
14. Return smallest_time of Nc

Run time Complexity: O(n)

3.2. Architecture Design

The proposed architecture is further divided into three parts

3.2.1. Blockchain over Privacy Managers

This scheme aims to improve the unpredictability of pseudonym mixtures. It also reduces the cost and effort of constantly generating new pseudonym certificates as it provides reusing the pseudonyms after shuffling them. The social advantage in terms of expedient travel, users are recommended to take part in the blockchain platform to share and improve traffic and navigation information. Through blockchain traffic routes can be optimized and speed can be controlled hence reducing pollution of urban traffic.

Algorithm 4 explains the updated step-wise procedure as discussed by [129]. Nomenclatures and symbols used in algorithms are listed in **Table 3.1.**

Table 3.1. Notations

Notation	Description
v_i	ith vehicle in the ITS environment.
rsu_j	jth RSU in ITS environment and is the subset of RSU
$CERT_i$	It is the subset of $CERT$ and is the ith certificate of the ith vehicle.
PKI	Public Key Infrastructure
PID_i	It is the subset of PID and represents pseudonym identities [1,2,…., N]
pid_i	Permanent IDs
pk_i, sk_i	Public keys and Private (Secret) keys of permanent as well as pseudonyms IDs.
V_n	All the vehicles of the network
RSU_i	ith RSU
PM	Privacy Manager
PMi	ith Privacy Manager
tx	Transactions

- PKI will be used for the initial registration of vehicles i.e it generates and broadcasts the ID, PK, SK, and CERT to manufacturers via a secure channel using cable connection or fiber optic.

- Manufacturers will give the above credentials to vehicles.
- PKI then generates and broadcasts PS_i, certificates, pk, and sk to PM_i encrypted with the public key of PM and signature encrypted with the secret key of the PKI.
- PM_i and RSU will broadcast the PS_i to RSU and vehicles respectively.
- Vehicles will use the PS_i sets and returns the used or unused pseudonyms to RSU after its expiration date.
- Privacy Managers (PMs) will collect used pseudonyms sets in form of packages from RSUs. Further PMs Upload the pseudonym sets to PM cloud.
- PMs would shuffle them in the cloud and would take the shuffled sets back according to the traffic need reported by RSUs.
- Each PM would now do mining on the sets they have collected from the cloud. The mining process will be done according to the proposed consensus mechanism Proof of Pseudonym.
- The winner PM's sets of pseudonyms will be published into the block of blockchain and RSUs will now assign them to vehicles after it takes the sets from PMs.
- The proposed architecture for the pseudonym shuffling mechanism is depicted in **Figure 3.4.** All the above discussion can be seen step-wise in **Figure 3.4.**

3.2.1.1. Formal Model:

We formally model the proposed architecture and its important components as follows

DEFINITIONS

1) Vehicle

$V \supseteq \cup_{i=0}^{\infty} v_i$ V is a superset of all the vehicles in the ITS environment.

2) RSU

$RSU \supseteq \bigcup_{i=0}^{\infty} rsu_j$, RSU is a superset of all the Road side units in the ITS environment.

3) Certificate Set

$CERT \supseteq \bigcup_{i=0}^{\infty} cert_i, CERT$ is a superset of all the certificates. Certificate $cert_i$ is the ith certificate.

4) Certificate to Vehicle Assignment

$CAV \supseteq \bigcup_{i=0,j=0}^{\infty} PKI \xrightarrow{cert_j} v_i$, CAV contains all the assignment of certificates $cert_j$ to vehicles v_i.

5) Pseudonymous IDs

$PID \supseteq \bigcup_{i=0}^{\infty} PID_i, PID$ is a super set of all the Pseudonymous IDs in a domain.

6) PAV

$PAV \supseteq \bigcup_{i=0,j=0}^{\infty} PKI \xrightarrow{PID_i} vi, PAV$ contains all the assignments of Pseudonymous IDs (PID_i) to vehicles v_i.

7) PM

$PM \supseteq \bigcup_{i=0}^{\infty} rsu_j, PM$ is a superset of all the Privacy Managers in the ITS environment.

Algorithm 4 "Pseudonym Shuffling"

---

1. **Initial Registration:**
2. PKI generate constant

$pid_i, cert_i, pk_i, sk_i$

3. PKI send

$pid_i, cert_i, pk_i, sk_i$

through the *secured channel* to the Manufacturer.

---

4. Manufacturer issue

   $pid_i, cert_i, pk_i, sk_i \leftarrow V_n$

5. PKI generate

   $pid_i, cert_i, pk_i, sk_i$

6. PKI broadcast

   $\{pid_i, cert_i, pk_i\}_{skpk(PM),}\ signature_{sk(PKI)}\ to\ PM_i$

Pseudonym Allotment:

7. $PID\ broadcast\ pid_i, RSUi\ to\ V_n$

Pseudonym Management:

8. **for** (y = 1; y≤ i ; y++) **do**
9. PMi collects all the used (expired) pseudonyms i.e.
10. $PID \supseteq \bigcup_{i=0}^{\infty} PID_i$ from RSU_i
11. Counts the number of used PID = n ;
12. Encapsulates PID into a package →PM cloud network.
13. PMi picks PID sets in PM cloud.
14. PMi shuffles the PID sets in PM cloud and relocate them to destination PMi.
15. **end for**
16. PMi starts mining on the selected PID sets.
17. The block is broadcasted into the network by the winner miner.
18. **for** (y = 1; y≤ i ; y++) **do**
19. PID retrieves new pseudonyms
20. Repeat steps 2 and 3

21. **end for**

22. Exit

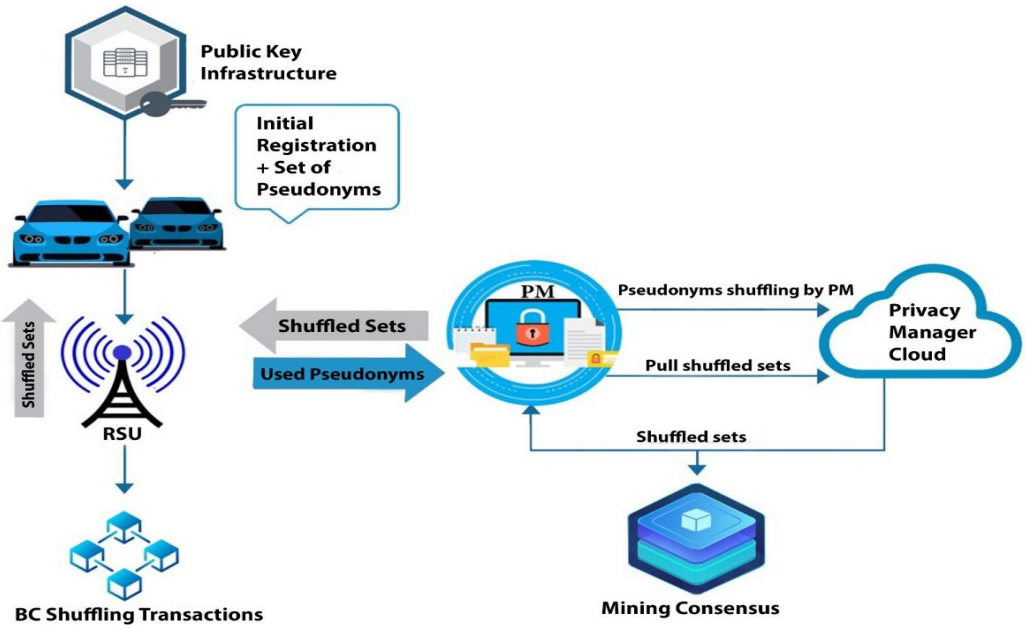

Figure 3.4. Architecture Design focusing the Pseudonym Shuffling in Blockchain over PMs

3.2.2. Format of the Transaction

The transaction ledger encapsulates pseudonyms and their credentials from a source to a destination privacy manager. **Table 3.2.** shows the ledger containing transactions. In the record, source and destination PM addresses are mentioned on the left side, and the right side contains the pseudonym sets belonging to this transaction with corresponding IDs i.e. certificate and key pairs. The data of the payload on the right side of the table is encrypted using the public key of the destination Privacy Manager (PM_destination) pk. The data is encrypted in each column which institutes the transaction's confidentiality and integrity. The destination PM having the private key (PM_destination)sk can see this information by decrypting it. Another encryption is done for the messages with the source PM's private key (PM_source) sk. Using signa-

tures is important as it prevents the information of the transaction from eavesdropping and spoofing attacks. As they would need to counterfeit signatures. **Table 3.3.** shows the transaction format in the ledger, that contains the header of the transaction and payload. The number of transactions in the header specifies the order number of this transaction in the ledger. The source and destination addresses of PM are the same as Bitcoin inputs and outputs [43]. The signature placed at the end of the transaction is for the non-repudiation, integrity, and authentication of information related to the key transfer. The cipherinfo is the pseudonym that sets information encrypted with the public key of the target PM. The used pseudonyms are contained in the payload.

Table 3.2. Transaction Ledger

Source	Destination	Sequence number	Pseudonyms
PM(A)	PM(1)	1	$PID_{(AA)}^{PM(A)}$
…	…	…	…
PM(B)	PM(2)	2	$PID_{(BB)}^{PM(B)}$
…	…	n_i	…
PM(X)	PM(i)	$\sum_{k=1}^{i} nk-ni+1$	$PID_{(XX)}^{PM(X)}$
PM(X)	PM(i)	$\sum_{k=1}^{i} nk$	$PID_{(XX)}^{PM(X)}$

Table 3.3. Transaction Format

Header of the Transaction
The transaction hashed result
Sequence number of this transaction in the current block
Current/source address of the privacy manager: PM_source
Receiver/ Destination privacy manager address: PM_destination
Signature of the current transaction

Sig{ PM_destination +CipherInfo}$_{(PM\_source)sk}$
Body/Payload: (Encrypted information of the Transaction)

3.2.3. Format of the Block

A block is considered to store all transactions as records. A large chain is formed when these blocks are combined called the blockchain. The block format is given in Table 3.3. The first row is the block number, which is the block order number in the entire chain. The cryptographic hash of the previous block securely links the current block to its parent block. This makes it extremely difficult to change adjoining sub-chains with different data and to assure the validity of these changes to other nodes of the network. Merkle tree root is used for the integrity and security of transactions within a block. All the transactions in this block are mutually authenticated into the Merkle tree root, hence any change in the transactions would result in a different Merkle root value. To prevent tampering in a block timestamps are added to prove that the transactions are made in a particular block as in bitcoin. The consensus field contains the part of validation where the agreement is taken place among nodes. The working of Proof of Pseudonym is explained in detail in section. The payload field holds the above-mentioned transactions that are allocated randomly by the block creator.

Table 3.4. Format of Block

Header of the Block	
Field	Description
Version	Number of Block Version
Previous Block Hash	Hash of the previous block in the chain
Merkle Tree Root	Hash of the Merkle tree root
Timestamp	Time of the block creation

Consensus	The Proof-of-Pseudonym
Body(Payload) of the Block (Transactions)	
Transaction 1 _ _ _ Transaction N.	

3.2.4. Blockchain over Road Side Unit

Each Privacy Manager (PM) manages several RSUs under its coverage area. RSU has a risk of a single point of failure as several vehicles are managed by the RSU so the risk is high if it fails to work. RSUs are vulnerable to insider and outsider attacks so there is a need to secure the data on RSU. RSU can be protected from attacks as well as from a single point of failure if it is decentralized hence we introduce the blockchain over RSUs. However one may think of the overhead created by maintaining multiple blockchains since the environment is in real-time and vehicles need quick responses. The issue comes with using slow consensus methods. The efficient consensus mechanism that we designed i.e. Proof of Pseudonym resolves the issue by generating the winner of the blocks faster as compared to other well-known methods. RSUs under each PM manages the blockchain locally among each other. The working flow is as follows:

In our scheme, we assume the RSUs contain three sorts of data: The vehicles status information of their coverage area, the shuffled sets that it gets from the PMs to further distribute among vehicles, and the used pseudonyms given by vehicles. To disseminate the announcement that RSUs got the shuffled pseudonym sets and to protect the sets against attacks the information is protected in the blockchain.

Each RSU participates in a consensus to reach an agreement. Since Proof of Pseudonym gives fast results so the blockchain is maintained efficiently. Similarly, when the RSU collects the used pseudonyms from the vehicles it puts them as a

separate transaction in blockchain to inform other RSU about the sets being used. The used pseudonyms are collected along with vehicle status information and recorded as a transaction. RSUs now run the mining process again to publish a block in a blockchain to keep the record of used pseudonyms. Hence forming a Blockchain of Blockchains. This is advantageous in a way that attackers cannot try to reuse the used pseudonyms sets of the vehicles. **Algorithm 5** discusses the pseudonym distribution by RSU while maintaining the blockchain.

The pseudonyms in the sets are tracked by the blockchain and any vehicle using the same used sets is identified and reported to PM which further proceeds to inform CA to revoke its certificate. Similarly, the blockchain keeps using pseudonym sets as transactions. As the data in blockchain are kept immutable, anonymous, encrypted, and transparent so the attackers cannot take an advantage of data kept in BC at RSU. The proposed scheme already maintains the blockchain as the main entity where the shuffling process takes place over the PMs in the cloud.

1) However, we keep the services of the server optional in the case of blockchain over RSU. There are two possibilities in this case. RSU can take the services from the cloud server for the Proof of Pseudonym i.e. percentage of nodes and nodes selection.
2) RSU can eliminate the use of servers but in that case, the code needs to be re-adjusted for randomization of the percentage for nodes selection. **Figure 3.5.** depicts blockchain over RSU.

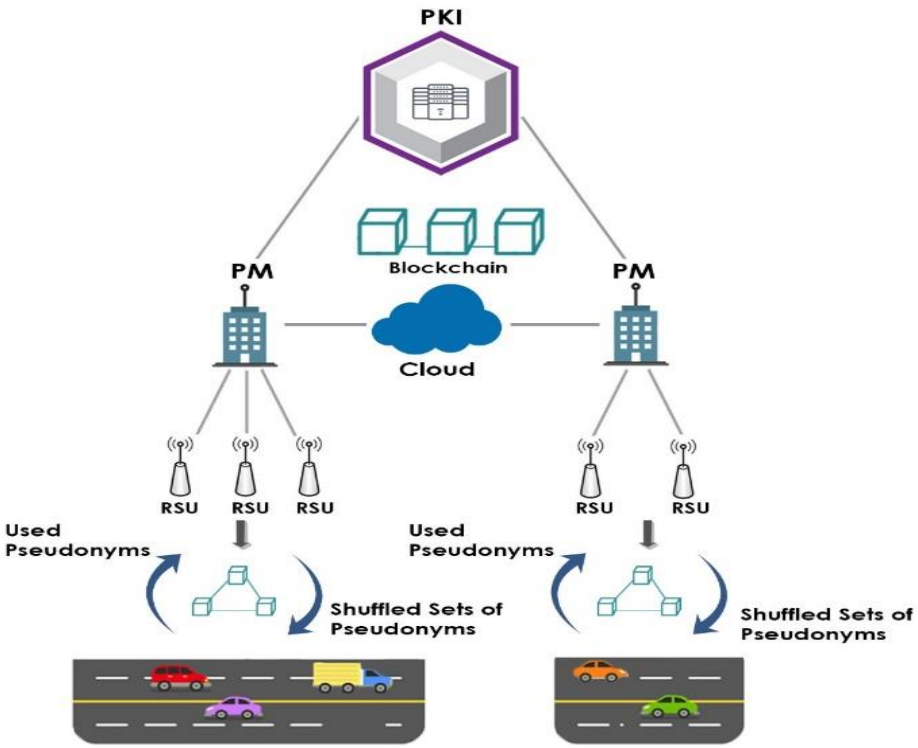

Figure 3.5. Architecture Design focusing the blockchain over RSUs

Algorithm 5 "Pseudonym Distribution of RSUs via Blockchain"

i. **Input:** $new\ PID$

ii. **for** (y = 1; y≤ i ; y++) **do**

iii. **if** {RSU_i requests for new PID } then

iv. PM_i sends the PID to RSU_i.

v. **end if**

vi. RSU_i records the PID as tx in BC and starts mining.

vii. After the block is published RSU_i distributes the PID to V_n .

viii. V_n returns PID to RSU_i after using.

ix. RSU_i starts mining again for used PID_i

x. RSU_i sends the used PID_i to PM_i.

xi. **Output:** $used\ PID_i$

xii. **End for**

xiii. Exit

3.2.5. Proof of Pseudonym

Proof of pseudonym has the idea of electing nodes which is common in Proof of Kernel work but it is not solving the puzzle as is done in Proof of Work or Proof of Kernel Work. Proof of Pseudonym adopted the best features from Proof of Kernel Work and Elapsed time, hence we can say Proof of Pseudonym is novel. The novelty can also be seen in its efficiency from the results it provides in terms of time and memory presented in chapter 4. Proof of Kernel Work is like a proof of work in which miners are given a puzzle to solve while Proof of Pseudonym elects miners to generate random time which is not the case with Proof of Kernel Work. The novelty of the research is also the scheme for shuffling pseudonyms and managing them via two blockchains.

Proof of Pseudonym working as client and cloud server is as follows.

a) The server detects new nodes (clients) connected to the network. For example, $N_1, N_2, N_3, \ldots, N_n$ are the nodes connected with the server.

b) The Server decides randomly the percentage of miners among the participant nodes not less than 50 percent. The percentage is set to 50 because the consensus agreement is more reliable if at least half or more of the network nodes participate. The percentage is calculated by the following formula

$$\frac{(N_1+N_2+N_3+\ldots+N_n)}{N_x} * (\text{threshold value of percentage}) = i\% \text{ nodes} \ldots\ldots(1)$$

where N_x is the total number of nodes.

c) The server is also responsible to decide particular nodes among the percentage of nodes to proceed. For instance, we say the server chooses N_2, N_3, N_5 ... N_i. So it guesses randomly the nodes with their IP addresses. These nodes are informed about mining while other nodes are not informed about the elected miner nodes. Those nodes will only get a message that they are not selected for mining. This is to protect the miner nodes from any kind of vulnerability.

d) These miners will reach the consensus to publish a block of transactions into the blockchain. The time limit is also decided by the server for the nodes.

e) The nodes run the code for generating random short time.

f) The node with the shortest time is the winner of the race. Optionally it can get an incentive from the network. The proposed scheme runs over the PMs and RSUs so there is no need for incentive. However, the Proof of Pseudonym is not limited to pseudonym shuffling. This node's block will be published into the blockchain network. In our scheme, the blockchain is maintained by PMs in the cloud so the server is also in the cloud for deciding the percentage of miners and nodes participating in the mining race as shown in **Figure 3.6.** We also describe the working of Proof of Pseudonym in **Algorithm 6** where in step 1 client $C_{(IP)}$ wants to handshake with server Ser. If a handshake is established then step 4 is performed else step 5 is performed where the server finishes the handshake with $C_{(IP)}$. In step 4 server checks the number of nodes Nn and threshold percentage. If Nn is less than or equal to the threshold percentage then it grants access according to a percentage threshold value set by the server randomly. If $C_{(IP)}$ is granted access the list is updated and step 2 is performed. In step 2 if the server grants access to $C_{(IP)}$ then step 3 is performed where if

$C_{(IP)}$ is selected for mining and time is generated for selected nodes and step 1 is performed otherwise step 4 is performed.

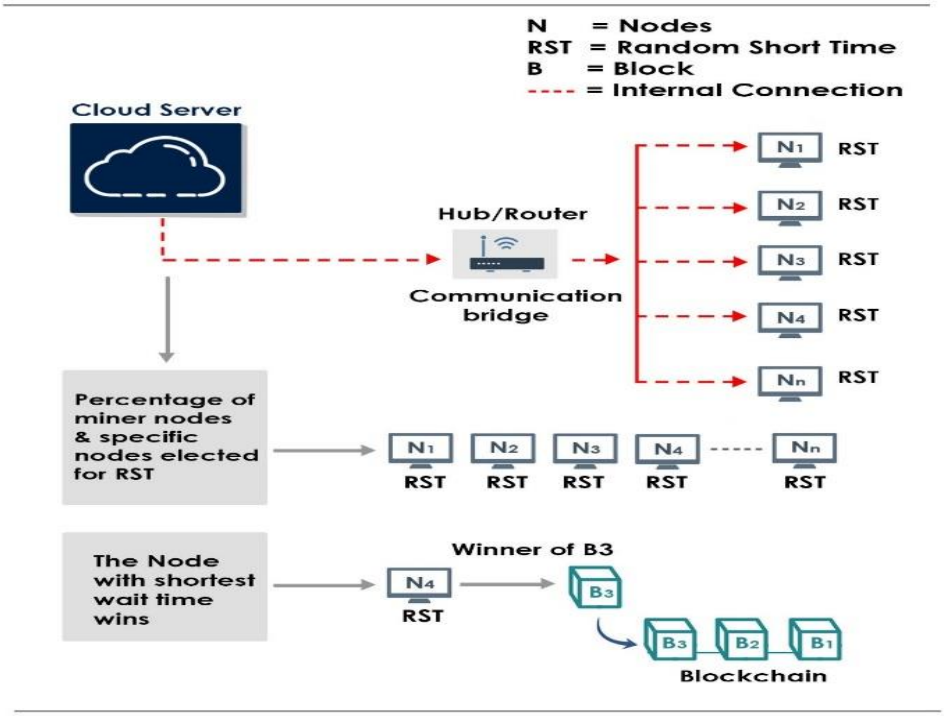

Figure 3.6. Proof of Pseudonym

Algorithm 6: Proof_of_Pseudonym_Consensus

1. **Inputs:**
2. $C_{(IP)}$: A client node that connects to the server using its IP address.
3. Ser: Server node
4. Port#: Port number for the connection
5. Nt: Node threshold value used by the server
6. Nn: Number of Nodes
7. **Begin Procedure**
8. Step-1. If $C_{(IP)}$ Requests Ser (handshaking)
   i. goto Step-4

ii. else

iii. goto step-5

9. Step-2. If $C_{(IP)}$ Request granted Ser

i. goto step 3

ii. else

iii. goto step 4

10. Step-3.

11. $C_{(IP)}$ is selected for mining

12. ts = $C_{(IP)}$ (Generate time for clients)

13. Goto Step-1

14. Step-4.

15. The server checks the Nn and Nt values

16. if Nn <= Nt

17. Nn = Access granted and list updated

18. Step-5. $\not{C}_{(IP)}$ Ser (Finish handshaking)

19. **End Procedure**

20. **Output:**

21. Either $C_{(IP)}$ is selected for mining or not.

Run time Complexity: O(n)

3.3. Use Case

The said scheme is a generic scheme to be deployed in ITS, however we discuss the use case of a battlefield having two scenarios a and b to understand the working of the scheme in particular scenarios. Therefore, our proposed scheme is not limited to only these two scenarios.

This use case describes the scenario of the battlefield where army vehicles communicate with base stations (RSUs) and shuffle pseudonyms frequently to avoid attacks on privacy. We discussed two scenarios of a battlefield.

a) We first describe the step-wise scenario of the legitimate vehicle and its pseudonym management on a battlefield as shown in **Figure 3.7**.

- In step 1 vehicle v1 encapsulates its pseudonyms in a package and sign it with its secret key and encrypt the package with base station public key i.e. $(PID1)_{sk(v1)pk(RSU1)}$.
- In step 2 the RSU records the pseudonyms in its blockchain as a transaction to keep it secure in a ledger and sends it to PM for shuffling with other pseudonym sets.
- In step 3 PMs shuffles the pseudonyms using cloud services. Each PM picks the sets randomly and allocates them to other PMs. All the PMs start mining using Proof of Pseudonym.
- In step 4 the winner PM pseudonym sets are validated and allocated to all the other PMs in the network.
- In step 5 the allocations are recorded as transactions in the network.
- In step 6 the RSU is given the sets according to its coverage area. The RSUs report the PM about the traffic of its domain.
- In step 7 the new shuffled sets are given to vehicles by the base stations.

b) Keeping in view attacks are not possible as the pseudonym shuffle takes place often but in the worst case, if an attacker attempts any kind of attack, it can be identified by PM via blockchain. In this case, we assume that the attacker uses the pseudonyms already given to other vehicles to confuse the network. We show this scenario in **Figure 3.8.**

- In step 1 when the attacker communicates with RSU where an RSU gets suspicious where it looks up in blockchain it manages.
- In step 2 RSU reports about this malicious vehicle to PM.
- In step 3 the PM again checks the blockchain transactions for the pseudonyms it contains. The blockchain transactions contain pseudonym sets with sender and receiver credentials that can be traced by RSU locally in its blockchain or by the PM.

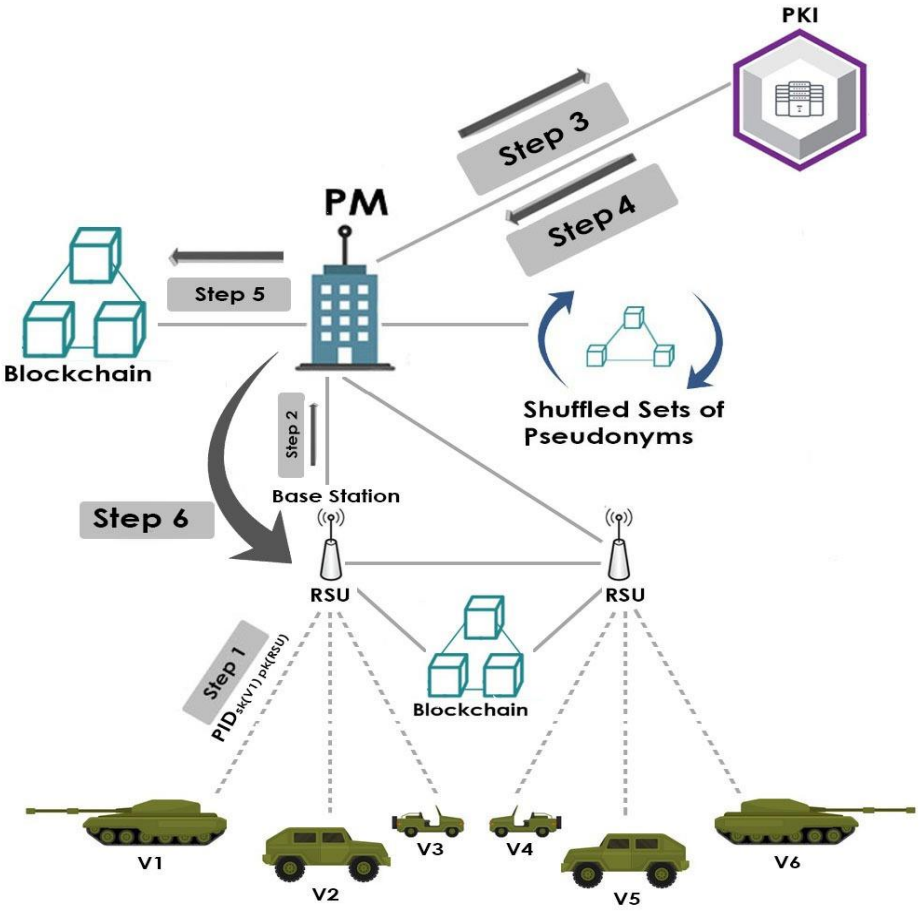

Figure 3.7. Proposed scheme on the battlefield.

- In step 4 the attacker's credentials are further reported to PKI
- In step 5 PKI revokes its certificate.

- In step 6 the information about revocation is sent to PM
- In step 7, PM adds the attacker to its blacklist and shares the information with the corresponding RSU.
- In step 8 RSU informs all the other vehicles about the malicious attacker and the vehicles do not communicate with it any further.

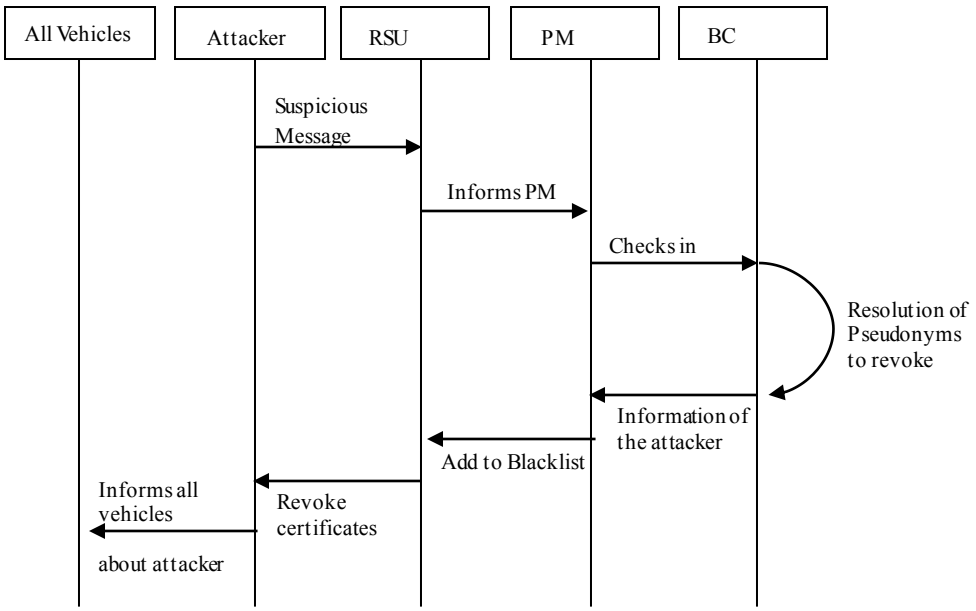

Figure 3.8. An attacker on a battlefield.

3.4. Summary

This chapter is all about the proposed architecture where the existing consensus mechanisms suggested for Intelligent Transportation Systems are discussed. The proposed architecture for pseudonym shuffling on blockchain technology is contributed with Proof of Pseudonym. RSU is protected from external and internal attacks via introducing a preliminary design of blockchain. In the end, we presented use cases with two sub-parts where the proposed scheme can be deployed in a real-world scenario.

Chapter 4: Simulation Results and Security Analysis

4.1. Development Platforms

We discuss the development platforms used by the consensus algorithms. **Table 4.1.** describes some of the important features of the development environments.

4.1.1. Ethereum

Ethereum is a cryptocurrency-based blockchain platform. It allows programmers to write smart contracts [132] which are self-executing methods. The language used for smart contracts writing is solidity. These smart contracts are executed by Ethereum Virtual Machine (EVM). Every Ethereum node must have an EVM that keeps the copy of the blockchain. The EVM uses a stack register of 256-bits which is designed to run the same code as expected. EVM is also referred to as Ethereum yellow paper and it has been implemented in C++, Golang, Java, JavaScript, Ruby, Python, and many others.

Table 4.1. Comparison of Blockchain Technologies

Comparison Parameters	Ethereum	Cosmos	Cardano	EOS	Bitcoin	Hyperledger	Corda
Token	ETH	ATOM	ADA	EOS	Bitcoin	n/a	SDK
Public / Private	Public	Public/ Private	Public	Public/ Private	Public	Public/ Private	Private
Programming Languages	Solidity	Java, C++, Python, Go	Haskell	JavaScript, Python, Ruby	Golang	Java, Golang, Node	Kotlin, Java
Consensus Algorithms	Proof of Work (Currently used), Proof of Stake (In Future)	Tendermint (Byzantine Fault-Tolerant, Proof of Stake)	Proof of Stake	Delegated Proof of Stake	Proof of Work	Practical Byzantine Fault Tolerance	Pluggable Consensus

Transactions Per Second	25	10,000	n/a	Millions (Theoretically)	1/3 to 1/7	More than 1000	Between 15 and 1678 TPS
Transaction Size	1 MB	250 bytes	n/a	n/a	1 MB	Changeable (depending on framework)	Maximum size in bytes
Open Source	True	True	True	True	True	True	True
Pros	Anyone can write a smart contract and anyone can view that contract	Works like a hub for blockchains, based on Tendermint	Scalability, Side chain reduce the risk of hacks.	Parallel processing, low latency, free usage (claimed not proven).	Safe and secure, High token value.	Don't use cryptocurrency so it is ideal for business networks.	Designed specifically for financial applications
Cons	Scalability issue, 25 transactions per second is very slow	Complex technology which may have compatibility issues with the latest technologies and new blockchains	Maintaining a side chain is complicated and it will require its miners.	Never actually free, not fully decentralized, the free transaction fee are imposed on everyone who has EOS.	Very slow, not ideal for programming while there are other faster technologies.	There are a lot of frameworks to choose from and they all have different requirements to implement and set up.	Partially decentralized, not much suitable for IoT resource-constrained networks

4.1.2. Cosmos

Cosmos is also called the internet of blockchains, which is a network for parallel blockchains. Before the idea of cosmos, blockchains were separated and isolated. Cosmos makes it easier for developers to build blockchains that can do transactions with each other. The end goal is the decentralized network of blockchains which is also referred to as "blockchain 3.0". It is open source and the Software Development Kit (SDK)[2]. Currently, the SDK is written in Golang but the cosmos team is open for changes. There are two other frameworks of cosmos technology which are Ethermint[3] and Lotion[4].

[2][Available online]: https://github.com/cosmos/sdk-application-tutorial
[3][Available online]: https://ethermint.zone/ and Lotion available online at: https://lotionjs.com/.
[4][Available online]: https://lotionjs.com/

Ethermint provides functionality like standard Ethereum including all the smart contracts and EVM the only difference is that Ethermint uses PoS instead of PoW. Lotion is a javascript-based tendermint consensus through which developers can make blockchains on the Cosmos network.

4.1.3. Cardano

It is a blockchain technology that allows the development of smart contract platforms. Cardano is made in Haskell programming language and it uses Plutus for smart contracts so both programming languages are functional. For decentralized architecture, the RINA network protocol is used for better bandwidth. In Cardano, a node does not have the entire blockchain it uses pruning and partitioning but it is not fully implemented yet.

4.1.4. EOS

EOS is a crypto currency powered by the EOSIO protocol [133]. This platform supports decentralized applications hosting decentralized storage and smart contracts. It uses the delegated proof of stake and is capable of running multithreaded which solves the issue of scalability better than most technologies. The main aim of EOSIO is to be a decentralized operating system that will allow developers to build decentralized applications such as steemit[5]. The token EOS provides storage and bandwidth of the network so the percentage of EOS owned shows the percentage of bandwidth available to the owner. At present only 21 block producers are allowed who can generate blocks in 500ms.

4.1.5. Bitcoin

Bitcoin[6] is the first cryptocurrency platform and it has the most expensive cryptocurrency. It has a public distributed ledger (the main blockchain) which keeps

records of all the transactions of bitcoin currency. Those transactions are verified through cryptography and this cryptography is done by nodes called miners. The miner who solves the cryptographic puzzle first gets the reward in the form of cryptocurrency. To solve the cryptographic puzzle first some miners join to create a mining pool [134].

Bitcoin[7] is solely a cryptocurrency platform, it also offers development documentation but those are focused on wallet managing. It is not as flexible as other platforms, it does not have any smart contract features or any application platform support.

4.1.6. Hyperledger

The Hyperledger blockchain technology is out of the box, unlike others. It does not have any cryptocurrency, it is just a technology that allows developers to build a whole new blockchain [135]. It is hosted by a Linux foundation designed to build private blockchains [136]. There are many frameworks and tools for hyperledger which are given in **Figure 4.1**.

4.1.1. Corda

Traditional blockchains may not be appropriate for many financial scenarios [136]. Corda provides the platform for smart contracts with some salient features i.e. It records and manages financial agreements between parties with compatibility of existing constructs. It supports several consensus mechanisms. Corda supports regulatory observer nodes. It allows access to data to only privileged ones [138]. The languages it supports are Kotlin and Java [139].

---

[5][Available online]: https://steemit.com/
[6][Available online]: Nakamoto, Satoshi. "Bitcoin: A peer-to-peer electronic cash system." (2008)
[7][Available online]: https://bitcoin.org/en/developer-documentation

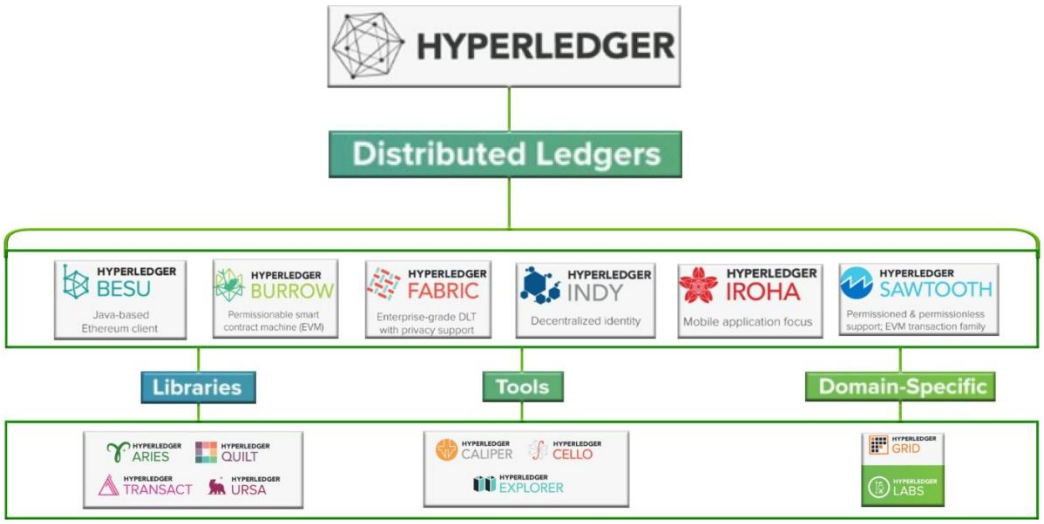

Figure 4.1. Hyperledger Framework and Tools

4.1.2. Limitations of Bitcoin, Ethereum, and Hyperledger

The following are the limitations of Bitcoin, Ethereum, and Hyperledger.

a. Privacy Issues:

Ethereum and Bitcoin both are public which means anyone can see the transactions participating in BC. However, areas like healthcare and financial service providers would want their transactions not to be seen by everyone in the network.

b. Scalability:

The public BC suffers scalability as reduces system throughput by allowing all the participants to do processing on the transactions.

c. High Storage Requirement:

All the BC nodes store all the data of the BC in the blocks. Consequently, storage requirements grow with time. The total number of transactions is restricted due to the limited size of the block.

d. High Power and Energy Requirements:

Bitcoin and Ethereum use PoW algorithm whereas Hyperledger uses Byzantine fault Tolerance consensus algorithm. Both require high computing power and energy. Also, Ethereum is more suitable for smart contract-based BC.

Due to the above drawbacks of Ethereum, Bitcoin, and Hyperledger, we implement the proposed framework using our private blockchain. The proposed framework has its assumptions and requirements like block structure. Furthermore, the proposed framework needs a consensus method that helps in the faster winning node of the block. Therefore, the proposed framework cannot be implemented in the existing BC platforms.

4.2. Results

The popular platform for blockchain development such as Hyperledger and Ethereum are not suitable for our proposed work due to their implementation of Proof of Stake (PoS) and Proof of Work (PoW) consensus algorithms. Moreover, their platform architecture does not support assumptions for developing customized consensus. The proposed architecture is implemented by developing a private blockchain in node.js and python platforms due to their rich support of libraries for the blockchain. To validate the algorithm emulation is performed by using mining rigs/GPU nodes. GPU RAM is 16GB, Processor: Intel (R) Core (TM) i7- 9700K CPU 3.60GHz 64 bit Operating System. We assume the constant specifications as the proofs are meant to run on PMs and not vehicles with variable specifications.

The algorithms are analyzed according to their work. The PoW solves a puzzle where the difficulty is chosen by the leading zeros. The more leading zeros result in increased difficulty levels.

4.2.1. Run Time Complexity

We also calculated the best, average, and worst time complexities of the three algorithms in **Table 4.2.** asymptotically. We can see that the Best and worst time complexity of PoW, PoET, and Proof of Pseudonym is the same and the average complexity of PoW and PoET is also the same i.e O(n) but Proof of Pseudonym is O(n/2). Hence the average time complexity of Proof of Pseudonym is better than the others, this is because not all the nodes are being involved in the process of mining and are not solving a complex puzzle that reduces its time. The average best and worst cases of the algorithms are calculated as follows:

a) Complexity of PoW 1 and PoW 2:

i. **Best Case:** The best case is calculated when the puzzle matches the string in both the algorithms in the least possible running time hence, deciding the winner of the block.

ii. **Average Case:** The average case is calculated when the puzzle matches the string in average running time.

iii. **Worst Case:** The worst case is calculated when the puzzle matches the string in maximum running time.

b) Complexity of PoET:

i. **Best Case:** The best case of PoET is achieved when the winner is found among the nodes in the least possible time.

ii. **Average Case:** The average case is when the winner is found among the nodes in average time.

iii. **Worst Case:** The worst case is achieved when the winner is found among the nodes in maximum time.

c) Complexity of Proof of Pseudonym:

i. Best Case: Best case o Proof of Pseudonym is calculated for the least time

taken by the algorithm to decide the winner with the shortest random time.

ii. **Average Case:** Average case of Proof of Pseudonym is calculated for the average time taken by the algorithm to decide the winner with the shortest random time.

iii. **Worst Case:** Worst case is when the winner is decided among the nodes in maximum time i.e. node having the shortest random time is reported in maximum time.

Table 4.2. Complexities of Algorithms

Run Time Complexity	Best	Average	Worst
Proof of Work 1	O(1)	O(n)	O(n)
Proof of Work 2	O(1)	O(n)	O(n)
Proof of Elapsed Time	O(1)	O(n)	O(n)
Proof of Pseudonym	O(1)	O(n/2)	O(n)

4.2.2. Pseudonym Shuffling Time Composition

We denote the number of PMs by N. To compute the execution time cost in a total of the shuffling process, we need the number of transactions ($n_T \in R$) where R is a set of all possible transactions. Given that,

$R = 2 *(n -1) *n$ .....(2)

where $n \in Z^*$ and $Z^* = \{0\}_u Z^+.Z^+$ represents the positive integers. The running time taken by Proof of Pseudonym is calculated and then we compare PoW 1, PoW 2 and Proof of Pseudonym according to the formula

$t_B = n_T * t_V + 2*t_P + t_P + t_M$……..(3) [129].

where t_B is the total time of the block. n_T denotes the number of transactions, t_V is the total time of transaction verification where t_V depends on n_T, t_P is the network cable propagation time, tprep is the block preparation time and t_M is the mining average time. Where mining time t_M is changed with the average time taken by PoW 1, PoW 2, and Proof of Pseudonym according to our analysis as shown in **Figure 4.6.** and **Figure 4.7**. We take constant time for t_V, t_P, and tprep with as negligible values as possible i.e. for t_V we take 0.001, for t_P, we take again 0.001 and we assume t_P to be zero to show the minimum possible running time an algorithm takes. Whereas we take variable values of n_T for 100 to 1000 as the transactions generated in off hours are 100 while in peak hours may exceed than 1000 as discussed in previous works.

1) Analysis of PoW 1

As there is no clear implementation available of Proof of Work in literature so we implemented it using two different algorithms to see the maximum and minimum time it takes to solve a puzzle. **Figure 4.2.** shows puzzle difficulty level on the x-axis and CPU time on the y-axis. The graph shows a clear rise in CPU time as the difficulty gets higher. The algorithm was executed for 10 different puzzles on each system. We encountered a long delay for the difficulty with the three leading zeros. **Figure 4.3.** shows the average pageable memory taken by the puzzles (nonces) in kilobytes in the CPU. The graph shows that as the puzzle difficulty level gets higher, the memory as well is occupied more. The GPU results in better time as compared to CPU but GPUs are expensive as compared to CPUs hence the resources used for PoW are expensive which is the drawback of PoW. This algorithm can even go to starvation as there is a chance that a puzzle can be guessed or it may happen that a puzzle cannot be guessed. We provide an improved version i.e. PoW 2 in the next section and compare it with

our proposed algorithm. We also show the total time of a block generated by different puzzles in **Table 4.3.**

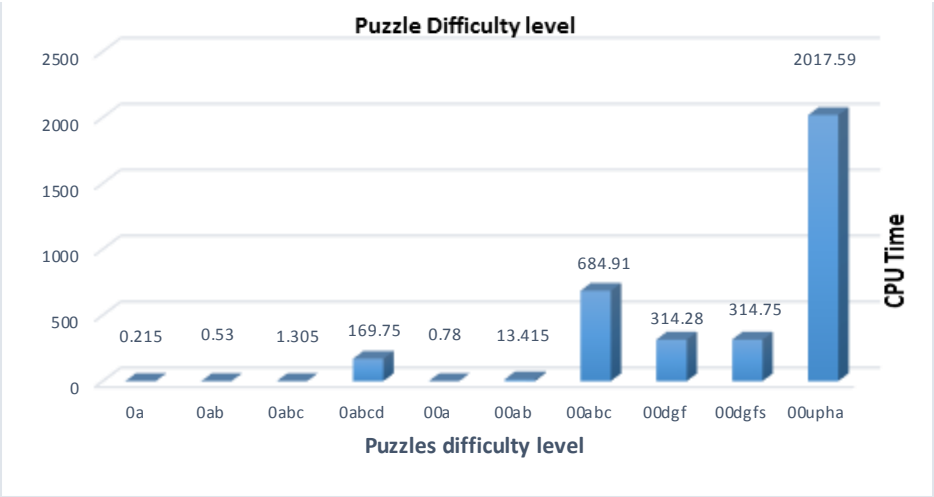

Figure 4.2. Proof of Work 1 puzzle-solving and the average time taken by CPU in seconds

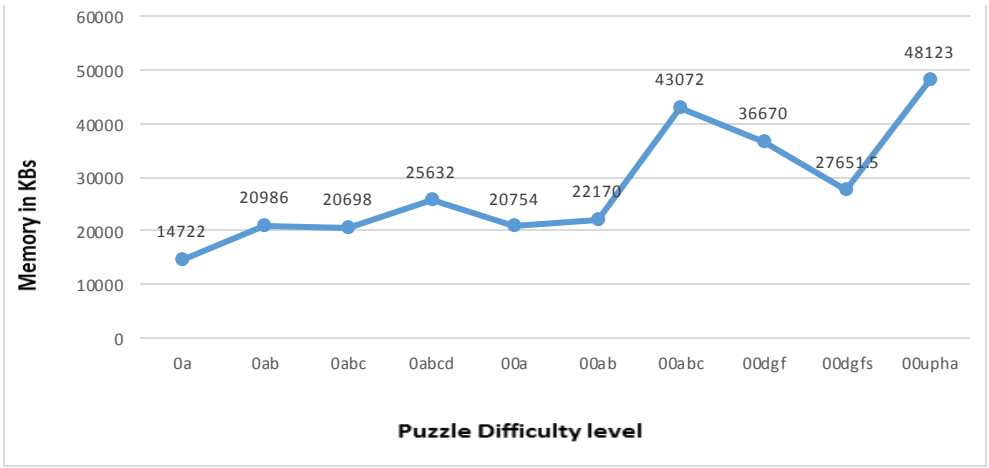

Figure 4.3. Proof of Work puzzle-solving and average pageable memory occupied by CPU in kilobytes

Table 4.3. Proof of Work 1 Time of Block.

S/no	Puzzle	Average CPU time	Transactions according to formula 3									
			100	200	300	400	500	600	700	800	900	1000
1	0a	0.215	0.337	0.437	0.537	0.637	….	….	….	….	….	1.237

2	0ab	0.53	0.652	0.472	0.852	0.952	….	….	….	….	….	1.552
3	0abc	1.305	1.427	1.527	1.627	1.727	….	….	….	….	….	2.327
4	0abcd	169.75	169.872	169.972	170.072	170.172	….	….	….	….	….	170.772
5	00a	0.78	0.902	1.002	1.102	1.202	….	….	….	….	….	1.802
6	00ab	13.415	13.537	13.637	13.737	13.837	….	….	….	….	….	14.437
7	00abc	684.91	685.032	685.132	685.232	685.332	….	….	….	….	….	685.932
8	00dgf	314.28	314.402	314.504	314.602	314.702	….	….	….	….	….	315.302
9	00dgfs	314.75	314.872	315.094	315.072	315.172	….	….	….	….	….	315.772
10	00uph a	2017.59	2017.12 2	2017.81 4	2017.91 2	2018.01 2	….	….	….	….	….	2018.61 2

2) Analysis of PoW 2

We also show the result of our second implementation of PoW 2 which generates a better running time than PoW 1 but its running time is unpredictable. We have calculated the running time of puzzle-solving and calculated the time of block with transactions from 100, 200….1000. We analyze the algorithm according to the above formula 3. It can be seen from the graphs in **Figure 4.4., 4.5. and 4.6.** This algorithm is more realistic in a way that it does not starve for long difficulty levels. We randomly present a few graphs with different difficulties i.e. leading zeros. The difficulty is set on the hash of data, timestamp, and nonce. So this algorithm is more close to the real implementation of PoW. However, there is the possibility that the variables such as t_V, t_P, and tprep can take more running time than we assumed. **Figure 4.4.** shows puzzle with difficulty 2 and it shows that the results shown in [129] are realistic but it has two disadvantages:

1) The execution time is dependent on a difficulty where the lowest difficulty generates faster results but at the cost of security as PoW with a low level is prone to 51% attack.

2) The execution time is unpredictable and is directly proportional to the level of difficulty. **Figure 4.5**. and **Figure 4.6**. presents PoW with difficulty 3 and 4 respectively with their respective CPU running times. Series 1 in **Figure 4.4**. and **Figure 4.6.** shows several transactions while series 2 shows CPU time taken by puzzle-solving in seconds.

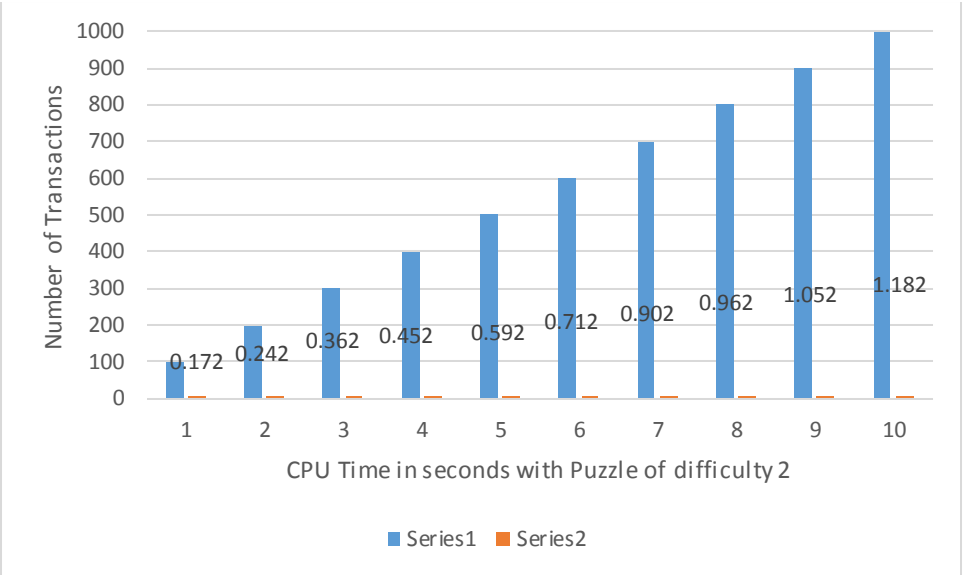

Figure 4.4. Proof of Work 2 (puzzle with difficulty 2)

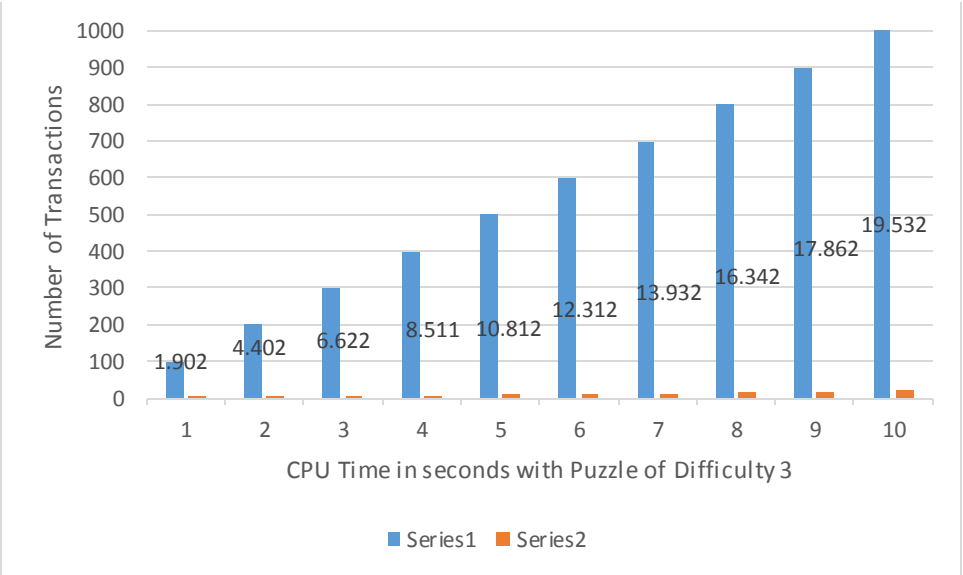

Figure 4.5. Proof of Work 2 (puzzle with difficulty 3)

PoKW is the delegated version of PoW so showing its graph is not mandatory. We

assume the execution time taken by puzzle in the network is not suitable for ITS applications as the vehicles need fast responses especially when it comes to privacy achievement. The removal or minimizing of the number of leading zeros makes the puzzle easy to guess hence more prone to 51% attack.

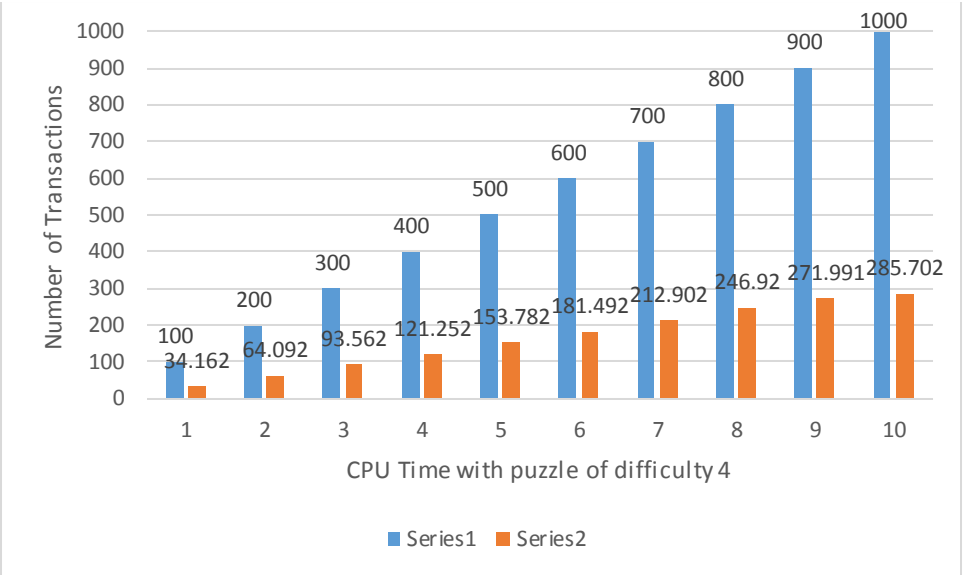

Figure 4.6. Proof of Work 2 (puzzle with difficulty 4)

3) Analysis of PoET

Figure 4.7. shows the results of PoET with 10 nodes and their randomly generated times. The said proof is tested for 100 and 200 virtual nodes but the graph can be shown clearly for not more than 10 nodes. We can see here that every node gets a chance to win and the random time generated is within 0.25s. Hence we say that this algorithm is better as compared to PoW and PoKW in terms of transaction winning time. The CPU time in PoET is directly proportional to the number of nodes. The drawback in this algorithm is that every node is the participant of the validation process hence the overall time it takes is more compared to Proof of Pseudonym.

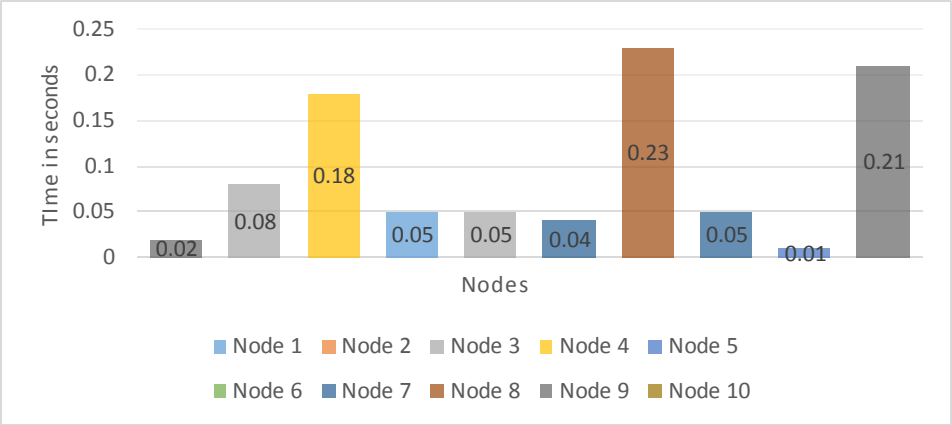

Figure 4.7. Proof of Elapsed Time winner node time

4) Analysis of Proof of Pseudonym

In contrast Proof of Pseudonym does not allow every node to participate in the process of validation. Hence minimizing delays as compared to all the above-discussed algorithms. The Proof of Pseudonym allows a certain number of nodes based on the random percentage decided by the cloud server between 10 to 50 percent. The rest of the nodes are notified if they are not elected for block validation but are not notified about the nodes elected for mining. This avoids any malicious attacks. Proof of Pseudonym does not contain solving any puzzle so it is not prone to 51% attacks. We discuss the security analysis in detail in section 4.3. We show the graph according to formula 3 in **Figure 4.8.** We elect 20 nodes for each shuffle consensus and we can see that all the nodes get a chance for generating the shortest random time and winning the race. We can see and compare the results of Proof of Pseudonym with both the implementations of PoW and PoET. Random selection can minimize the consensus time more. **Figure 4.9.** shows the comparison between the time generated by PoW 1, PoW2, and Proof of Pseudonym. We can see a clear drop-off time in Proof of Pseudonym as compared to both PoW. In **Figure 4.10.** we compare the time cost of transactions of Proof of Pseudonym with that of PoW used in a scheme of [129].

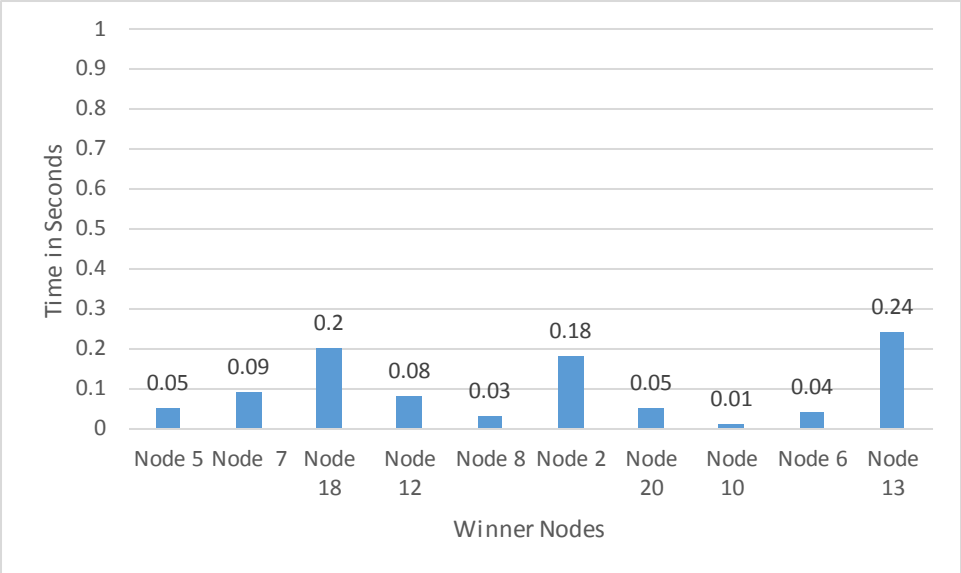

Figure 4.8. Proof of Pseudonym winner nodes time

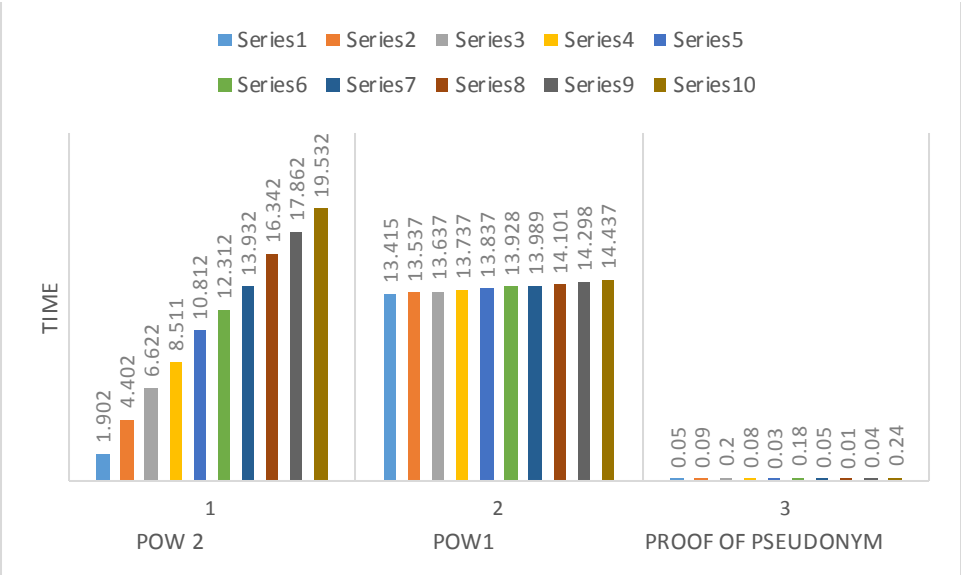

Figure 4.9. Comparison of Algorithms

4.3. Security and Privacy Analysis

We identify pertinent attack vectors for blockchains based on pseudonym management systems in vehicular Adhoc networks and mitigation measures for such attacks. We divide this section according to the privacy and security threats to pseudonym management schemes, the internal and external attacks on RSUs, and how they are prevented in the proposed scheme and last but not least we discuss the threats to consensus protocols we discussed i.e. PoW, PoET, PoKW and how we overcome

them in Proof of Pseudonym.

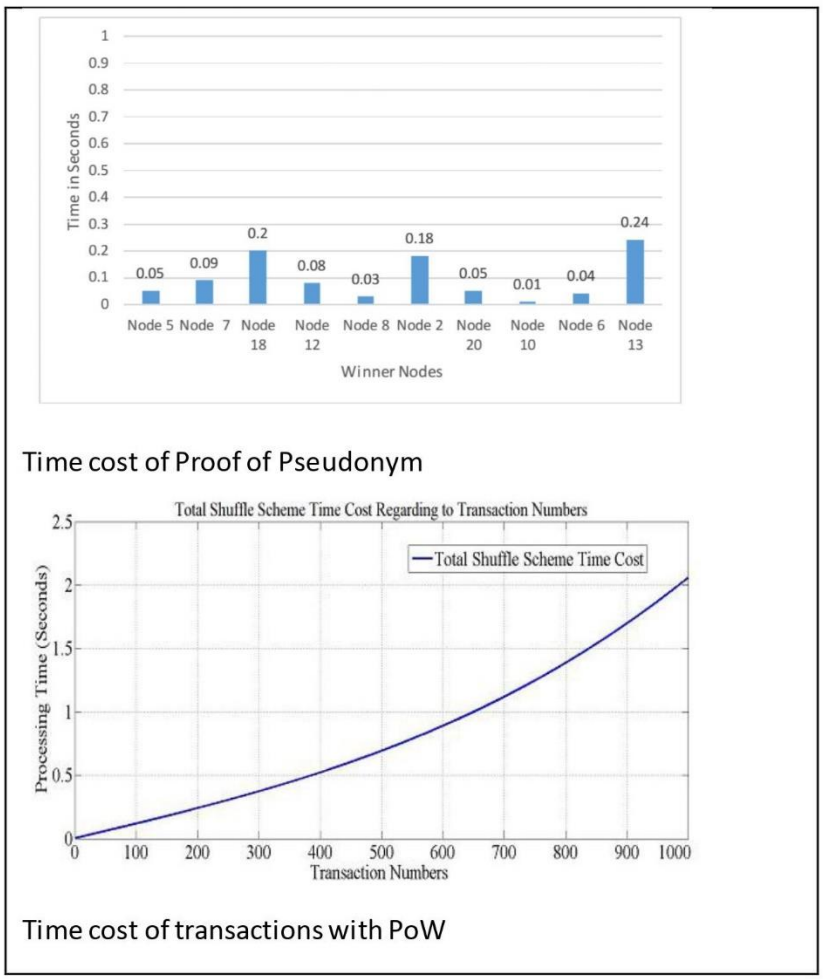

Figure 4.10. Comparison of our Proposed Consensus with [129]

a) No Single Point of Failure

The blockchain is distributed and the PMs and RSUs are all participating in the formation, maintenance, and updating of the blockchain of blockchains framework. Hence, no certifying authority is required and the single point of failure problem is removed. Additionally, the contributing nodes serve as multiple restoration and backup points.

b) The Identity Privacy

Unlike the traditional PKI, in which each certificate is linked with the name of the

public key holder. Our proposed framework does not save the identity of the owner in the blockchain. The generated pseudonyms are attached to the owner's account initially and later they are reused by other vehicles. The system can decide when to change and allot new pseudonyms. This way the privacy of the entities is achieved.

c) The Unlinkability

Unlinkability is the property that two messages sent by the same vehicle cannot be linked for a long time. In this proposed scheme the pseudonym given to the vehicles are not static and they keep on changing when the system triggers. Hence the messages sent by the same vehicle are unlinkable. Also by the use of blockchain the transactions of the vehicles pseudonym using is anonymous hence achieving unlinkabilty.

d) Non-Repudiation

Even though the framework is aimed to preserve the privacy of vehicles especially, since the blockchain is fully distributed, non-repudiation is also another essential property for assurance. Therefore, we have guaranteed that the mischievous nodes are responsible and that they cannot deny their committed activities.

The point that the pseudonym is present in the blockchain stops the vehicle from denying to have generated it. Also, if the vehicle has a pseudonym misbehaves, it gets revoked by the RSU near it or by Certificate Authority (CA) in case it is not under RSU coverage area.

4.3.1. Threats and Counter Measures for Shuffling Scheme

The safety messages are broadcasted in the network and an adversary can track the location of the vehicle via analysing these safety messages. We focus on external and internal attacks. External attacks are further divided into Global Passive Adversary

(GPA) and Local Passive Adversary (LPA). A Global Passive Adversary has inclusive access to a network of connected vehicles. An eavesdropper can leverage the broadcasted beacons messages and track a vehicle in a region where the eavesdropper is interested.

The Local Passive Adversary (LPA) is interested only in its transmission range region. It can also eavesdrop on the broadcasted beacons messages. Internal attacks have further two types Internal Betrayal Adversary and Internal Tricking Adversary.

Internal Betrayal Adversary (IBA) is a compromised node that can give information about vehicles by spoofing safety messages to a local or global passive adversary. The attacker can then find out the real identity and location of the target vehicle.

Internal Tricking Adversary (ITA). This adversary uses pseudonyms of other vehicles to confuse the network and attack a node.

We show how this study overcomes the attacks discussed above.

1) Global Passive Adversary and Local Passive Adversary

The most common type of attack is Global and Local passive adversaries' privacy attack where the vehicle's beacon messages are passively snooped. The GPA and LPA might gain the location and timestamps of the leaving and joining of vehicles to map them. Works in [38, 39] state that they can guess the vehicle's location-based data by brute-forcing beacon messages even if the vehicles change their pseudonyms often. Instead, this system allows vehicles to change the pseudonyms where there is maximum traffic in that zone. The GPA and LPA hence cannot map the timestamps and locations of the vehicles as there is no leaving or joining of vehicles at mixed zones.

2) Internal Betrayal Adversary and Internal Tricking Adversary

Internal Betrayal Adversary (IBA) can obtain a vehicle's pseudonym to map it with the vehicle's real identity or it can give it to a global adversary for manipulation. This attack is avoided in this scheme because the vehicles are not supposed to exchange pseudonyms among them instead they modify their sets with their own sets given by the RSU of their coverage area. Vehicles cannot evaluate the source of the pseudonyms once it gets updated sets from the RSUs. The IBA cannot get any useful information from its vehicles or even tries to get the pseudonym sets. RSUs will identify the malicious activity as it records the transaction over a blockchain.

However, the Internal Tricking Adversary (ITA) will try to use the pseudonyms already given to the vehicles to confuse the network. This study finds out a solution for this problem where this activity of using old pseudonyms is identified by the nearest RSU of the coverage area. The identified adversary is marked and other vehicles gets informed about the attacker. Just in case the attacker leaves its RSU coverage area even then RSU of the new coverage area can figure it out as the blockchain is maintained among various RSUs under each PM.

RSUs and PMs will only identify the transactions of their coverage area and no other unencrypted transactions. The pseudonym allocation cannot be identified by the RSU or PM and hence the adversary cannot be identified if it's using old/false pseudonyms. However, when it launches the attacks like spoofing messages, it can be identified by the RSU if the attacker is in its range otherwise PM will identify and report to CA. The CA will revoke the credentials of the attacker.

3) Spoofing Attack

The PMs can be spoofed for an attack where they can generate false blocks for pseudonym shuffles and can manipulate the permanent identity of the vehicles. The proof of pseudonym consensus requires that the nodes selected for mining are

informed and other nodes are not informed. So the possibility to attack a PM is very rare as the network does not know about the PMs selected for mining. In case PM is compromised, the CA will discard it and the network will discard its pseudonyms.

4.3.2. Attacks on Road Side Units and its Counter Measures

RSUs as a single point of authority for many vehicles suffer attacks. They can be compromised for mutual collaboration with an attacker or between each other. We discuss the possible attacks and their measures by our proposed system.

1. Internal Attacks

RSU can reveal a set of pseudonyms it gives to a vehicle to an attacker allowing an attacker to track the location of the vehicle. An attacker also can track vehicles between two RSUs in case both the RSUs are compromised. Curious or compromised RSU will try to analyze the communication habits between different real vehicles and the user's real identity behind the vehicle to track the vehicle services. In our proposed system the RSU contains pseudonym shuffled sets as well as the used pseudonym sets. Therefore, first, the RSU contains pseudonyms and that is too shuffled so the curious RSU cannot take advantage even if it observes the communication habits and cannot reach to map the original identity with the pseudonyms. Curious RSU cannot analyze the shuffled pseudonym sets as it is of no use. Secondly, the RSU contains used pseudonym sets, again this data is of no use to RSU. With blockchain, the transactions are stored anonymously. So even the RSU cannot know the pseudonyms coming from a particular vehicle.

2. External Attacks

External attackers can attack RSU by stealing the data stored in RSU. An attacker may forge a large amount of communication data within RSU and may perform other

illegal operations. Hence RSUs are at risk. With blockchain over RSU, the transactions and data are secured in a distributed ledger that is immutable. External attackers cannot have access to data stored in RSU and if they by any means have access to RSU data, it is of no use to them as the transactions are anonymous.

4.3.3. Attacks and Counter Measures in Proof of Pseudonym

Proof of Pseudonym prevents the possible attacks encountered in PoW, PoKW, and PoET. It prevents 51% of attacks as it does not involve computational resources. It is not hardware independent as we analyzed it on core i3 and core i7 systems. Proof of Pseudonym also prevents Sybil attacks as it chooses selected nodes for consensus and not all. The selected nodes are not aware of each other about their selection for consensus hence it prevents Sybil attack. As this algorithm does not involve solving a puzzle so an attacker cannot separately mine a private thread.

4.1. Summary

This chapter is all about the results which we collected for the CPU time for PoW to show the minimum time it takes for the simplest puzzle. We also show how much memory is consumed by PoW using different puzzles. Security analysis of the proposed architecture is carried which is pseudonym shuffling over PMs, Blockchain over RSU and then we show how Proof of Pseudonym avoids attacks. We also provide formal modeling for the mentioned algorithms.

Chapter 5: Conclusion

5.1. Conclusion

Intelligent transportation broadcasts the beacon messages among vehicles using their real identities. These broadcasted messages can be utilized by adversaries for launching attacks. To facilitate the users to keep privacy protected, the use of pseudonyms instead of using real identities is the solution. This thesis proposes a scheme for pseudonym management in intelligent transport systems based on blockchain. Pseudonyms hide the real identity of the user to preserve user identity. Pseudonym generation creates overhead in the network so reusing the used pseudonym by shuffling them, again and again, is the optimal solution. Shuffling was done by a third party that is prone to attacks and is a single point of failure. The blockchain-based shuffling scheme for pseudonym management scheme suffered delays scheme due to the use of the Proof of Work consensus method. So we proposed a novel consensus method Proof of Pseudonym which overcomes the delays suffered by Proof of Work and the other two suggested algorithms for the blockchain-based shuffling scheme are Proof of Kernel Work and Proof of Elapsed Time. First, the scheme uses PMs to collect the used pseudonyms from RSUs which are ultimately collected from vehicles. The pseudonyms are shuffled using cloud server services and the sets are then collected by PMs and distributed back to RSUs. Second, the said scheme is complemented with a novel Proof of Pseudonym algorithm which shows its efficiency from the results. The consensus is designed where the server decides the percentage of participating nodes as well as the particular nodes selected for mining. The server decides the nodes using their IP addresses. The result shows that the proposed consensus achieves better results in terms of execution time, memory, and

transaction processing time. Last but not the least, the idea of blockchain is introduced on RSUs under each PMs to keep the shuffled sets as well as the used pseudonyms collected from vehicles protected from internal and external attacks.

5.2. Limitations

This research has the following limitations:

a. RSU with blockchain is a theoretical concept and may pose an overhead in the network as the validation process is achieved twice. RSU performs consensus for collecting used pseudonyms from vehicles and the second time it performs consensus when RSU gets shuffled sets from PMs.

b. Transactions of the RSU blockchain need to be refined.

c. Proof of Pseudonym needs to be revised for blockchain over RSU.

5.3. Future work

In this thesis, we have presented a novel solution for pseudonym management using the Proof of Pseudonym suggested in this work. The said scheme achieves better transparency by the use of blockchain. The efficiency of the blockchain-based for pseudonym management is achieved using the Proof of Pseudonym consensus algorithm.

In the future, we plan to implement the blockchain over RSU and analyze the overhead as the mining process commences twice as per our assumption. The process of block validation is performed by the RSUs when it collects the used pseudonyms and again when it distributes the shuffled sets. There is a need to design the framework of weather to commence the validation twice or once. The overhead can be analyzed in terms of how efficient the scheme is, with validating the blocks. Proof of Pseudonym

over RSU blockchain needs to be updated for server services. It can be updated so it chooses the percentage of nodes without relying on the server or the blockchain over RSU may eliminate the process of consensus as the blockchain is managed just to secure the pseudonym sets from tampering. The transactions of the block in RSU need further consideration. Additionally, the Proof of Pseudonym can be edited with incentivizing process if used in other scenarios of ITS.

Bibliography

[1] N. Chaudhry and M. M. Yousaf, "Consensus algorithms in blockchain: Comparative analysis, challenges and opportunities," in *2018 12th International Conference on Open Source Systems and Technologies (ICOSST)*, 2018, pp. 54-63.

[2] A. Tapscott and D. Tapscott, "How blockchain is changing finance," *Harvard Business Review,* vol. 1, no. 9, pp. 2-5, 2017.

[3] X. Xu, I. Weber, and M. Staples, *Architecture for blockchain applications*: Springer, 2019.

[4] D. Puthal, N. Malik, S. P. Mohanty, E. Kougianos, and C. Yang, "The blockchain as a decentralized security framework [future directions]," *IEEE Consumer Electronics Magazine,* vol. 7, no. 2, pp. 18-21, 2018.

[5] P. Chatzigiannis, F. Baldimtsi, I. Griva, and J. Li, "Diversification across mining pools: Optimal mining strategies under pow," *arXiv preprint arXiv:1905.04624*, no., 2019.

[6] F. Hofmann, S. Wurster, E. Ron, and M. Böhmecke-Schwafert, "The immutability concept of blockchains and benefits of early standardization," in *2017 ITU Kaleidoscope: Challenges for a Data-Driven Society (ITU K)*, 2017, pp. 1-8.

[7] E. Landerreche and M. Stevens, "On immutability of blockchains," in *Proceedings of 1st ERCIM Blockchain Workshop 2018*, 2018.

[8] T. Aste, P. Tasca, and T. Di Matteo, "Blockchain technologies: The foreseeable impact on society and industry," *computer,* vol. 50, no. 9, pp. 18-28, 2017.

[9] A. Kosba, A. Miller, E. Shi, Z. Wen, and C. Papamanthou, "Hawk: The blockchain model of cryptography and privacy-preserving smart contracts," in *2016 IEEE symposium on security and privacy (SP)*, 2016, pp. 839-858.

[10] V. Buterin, "A next-generation smart contract and decentralized application platform," *white paper,* vol. 3, no. 37, 2014.

[11] Z. Zheng, S. Xie, H. Dai, X. Chen, and H. Wang, "An overview of blockchain technology: Architecture, consensus, and future trends," in *2017 IEEE international congress on big data (BigData congress)*, 2017, pp. 557-564.

[12] J. Golosova and A. Romanovs, "Overview of the blockchain technology cases," in *2018 59th International Scientific Conference on Information Technology and Management Science of Riga Technical University (ITMS)*, 2018, pp. 1-6.

[13] M. Nofer, P. Gomber, O. Hinz, and D. Schiereck, "Blockchain," *Business & Information Systems Engineering,* vol. 59, no. 3, pp. 183-187, 2017.

[14] R. Beck, "Beyond bitcoin: The rise of blockchain world," *Computer,* vol. 51, no. 2, pp. 54-58, 2018.

[15] C. Gañán, J. L. Muñoz, O. Esparza, J. Mata-Díaz, and J. Alins, "EPA: An efficient and privacy-aware revocation mechanism for vehicular ad hoc networks," *Pervasive and Mobile Computing,* vol. 21, no., pp. 75-91, 2015.

[16] F. Velde, "Bitcoin: A primer," no., 2013.

[17] A. Shoker, "Sustainable blockchain through proof of exercise," in *2017 IEEE 16th International Symposium on Network Computing and Applications (NCA)*, 2017, pp. 1-9.

[18] J. Brito and A. Castillo, "Bitcoin: A Primer for Policymakers, Mercatus Center," *George Mason University. Retrieved,* vol. 22, no., 2013.

[19] N. El Ioini and C. Pahl, "A review of distributed ledger technologies," in *OTM Confederated International Conferences" On the Move to Meaningful Internet Systems"*, 2018, pp. 277-288.

[20] E. Anceaume, A. Guellier, R. Ludinard, and B. Sericola, "Sycomore: a permissionless distributed ledger that self-adapts to transactions demand," in *2018 IEEE 17th International Symposium on Network Computing and Applications (NCA)*, 2018, pp. 1-8.

[21] A. Hughes, A. Park, J. Kietzmann, and C. Archer-Brown, "Beyond Bitcoin: What blockchain and distributed ledger technologies mean for firms," *Business Horizons,* vol. 62, no. 3, pp. 273-281, 2019.

[22] F. Rizal Batubara, J. Ubacht, and M. Janssen, "Unraveling Transparency and Accountability in Blockchain,"2019, pp. 204-213.

[23] G. Karame, "On the security and scalability of bitcoin's blockchain," in *Proceedings of the 2016 ACM SIGSAC conference on computer and communications security*, 2016, pp. 1861-1862.

[24] R. P. Wattenhofer, *Distributed Ledger Technology: The Science of the Blockchain*: Inverted Forest Publishing, 2017.

[25] W. Zhao, S. Yang, and X. Luo, "On consensus in public blockchains," in *Proceedings of the 2019 International Conference on Blockchain Technology*, 2019, pp. 1-5.

[26] C. Mohan, "State of public and private blockchains: Myths and reality," in *Proceedings of the 2019 International Conference on Management of Data*, 2019, pp. 404-411.

[27] S. Shahriar Hazari and Q. H. Mahmoud, "Improving Transaction Speed and Scalability of Blockchain Systems via Parallel Proof of Work," *Future Internet,* vol. 12, no. 8, p. 125, 2020.

[28] A. Ranade and Z. Shaikh, "A Survey on Blockchain Technology with Uuse-cases in Governance," *Available at SSRN 3568629*, no., 2020.

[29] S. Zhu, Z. Cai, H. Hu, Y. Li, and W. Li, "zkCrowd: a hybrid blockchain-based crowdsourcing platform," *IEEE Transactions on Industrial Informatics,* vol. 16, no. 6, pp. 4196-4205, 2019.

[30] J.-H. Lee, "BIDaaS: Blockchain based ID as a service," *IEEE Access,* vol. 6, no., pp. 2274-2278, 2017.

[31] G. Zyskind and O. Nathan, "Decentralizing privacy: Using blockchain to protect personal data," in *2015 IEEE Security and Privacy Workshops*, 2015, pp. 180-184.

[32] M. Sharples and J. Domingue, "The blockchain and kudos: A distributed system for educational record, reputation and reward," in *European conference on technology enhanced learning*, 2016, pp. 490-496.

[33] D. Parker, "Blockchain Voting Used By Danish Political Party," *CryptoCoinsNews. April,* vol. 23, no., 2014.

[34] S. Saroop, "Block chain Technology: Assessment from Application Perspectives," EasyChair 2516-2314, 2020.

[35] F. Þ. Hjálmarsson, G. K. Hreiðarsson, M. Hamdaqa, and G. Hjálmtýsson, "Blockchain-based e-voting system," in *2018 IEEE 11th International Conference on Cloud Computing (CLOUD)*, 2018, pp. 983-986.

[36] S. Wang, L. Ouyang, Y. Yuan, X. Ni, X. Han, and F.-Y. Wang, "Blockchain-enabled smart contracts: architecture, applications, and future trends," *IEEE Transactions on Systems, Man, and Cybernetics: Systems,* vol. 49, no. 11, pp. 2266-2277, 2019.

[37] G. Ali, N. Ahmad, Y. Cao, M. Asif, H. Cruickshank, and Q. E. Ali, "Blockchain based permission delegation and access control in Internet of Things (BACI)," *Computers & Security,* vol. 86, no., pp. 318-334, 2019.

[38] G. Ali, N. Ahmad, Y. Cao, Q. E. Ali, F. Azim, and H. Cruickshank, "BCON: Blockchain based access CONtrol across multiple conflict of interest domains," *Journal of Network and Computer Applications,* vol. 147, no., p. 102440, 2019.

[39] K. Christidis and M. Devetsikiotis, "Blockchains and smart contracts for the internet of things," *Ieee Access,* vol. 4, no., pp. 2292-2303, 2016.

[40] A. Chakravorty and C. Rong, "Ushare: user controlled social media based on blockchain," in *Proceedings of the 11th international conference on ubiquitous information management and communication*, 2017, pp. 1-6.

[41] J. Basden and M. Cottrell, "How utilities are using blockchain to modernize the grid," *Harvard Business Review,* vol. 23, no., pp. 1-8, 2017.

[42] V. Gramoli, "From blockchain consensus back to Byzantine consensus," *Future Generation Computer Systems,* vol. 107, no., pp. 760-769, 2020.

[43] S. Nakamoto, "Bitcoin: A peer-to-peer electronic cash system," Manubot2019.

[44] M. Osadchuk and R. Oliynykov, "Method of Proof of Work consensus algorithms comparison," *Радиотехника*, no. 198, pp. 105-112, 2019.

[45] D. Larimer, "Delegated proof-of-stake (DPOS). Bitshare whitepaper (2014)," ed: ed, 2014.

[46] L. S. Sankar, M. Sindhu, and M. Sethumadhavan, "Survey of consensus protocols on blockchain applications," in *2017 4th International Conference on Advanced Computing and Communication Systems (ICACCS)*, 2017, pp. 1-5.

[47] F. Muratov, A. Lebedev, N. Iushkevich, B. Nasrulin, and M. Takemiya, "YAC: BFT consensus algorithm for blockchain," *arXiv preprint arXiv:1809.00554*, no., 2018.

[48] I. Abraham and D. Malkhi, "The blockchain consensus layer and BFT," *Bulletin of EATCS,* vol. 3, no. 123, 2017.

[49] A. G. DAG, "Directed acyclic graph," no., 2013.

[50] S. M. H. Bamakan, A. Motavali, and A. B. Bondarti, "A survey of blockchain consensus algorithms performance evaluation criteria," *Expert Systems with Applications*, no., p. 113385, 2020.

[51] N. Chalaemwongwan and W. Kurutach, "Notice of Violation of IEEE Publication Principles: State of the art and challenges facing consensus protocols on blockchain," in *2018 International Conference on Information Networking (ICOIN)*, 2018, pp. 957-962.

[52] L. M. Bach, B. Mihaljevic, and M. Zagar, "Comparative analysis of blockchain consensus algorithms," in *2018 41st International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)*, 2018, pp. 1545-1550.

[53] M. Milutinovic, W. He, H. Wu, and M. Kanwal, "Proof of luck: An efficient blockchain consensus protocol," in *proceedings of the 1st Workshop on System Software for Trusted Execution*, 2016, pp. 1-6.

[54] A. Miller, A. Juels, E. Shi, B. Parno, and J. Katz, "Permacoin: Repurposing bitcoin work for data preservation," in *2014 IEEE Symposium on Security and Privacy*, 2014, pp. 475-490.

[55] A. Corso, "Performance analysis of proof-of-elapsed-time (poet) consensus in the sawtooth blockchain framework," no., 2019.

[56] M. S. Ferdous, M. J. M. Chowdhury, M. A. Hoque, and A. Colman, "Blockchain Consensuses Algorithms: A Survey," *arXiv preprint arXiv:2001.07091*, no., 2020.

[57] S. De Angelis, L. Aniello, R. Baldoni, F. Lombardi, A. Margheri, and V. Sassone, "PBFT vs proof-of-authority: Applying the CAP theorem to permissioned blockchain," no., 2018.

[58] M. Macdonald, L. Liu-Thorrold, and R. Julien, "The blockchain: a comparison of platforms and their uses beyond bitcoin," *COMS4507-Adv. Computer and Network Security*, no., 2017.

[59] L. Ren, "Proof of stake velocity: Building the social currency of the digital age," *Self-published white paper*, no., 2014.

[60] J. Innerbichler and V. Damjanovic-Behrendt, "Federated byzantine agreement to ensure trustworthiness of digital manufacturing platforms," in *Proceedings of the 1st Workshop on Cryptocurrencies and Blockchains for Distributed Systems*, 2018, pp. 111-116.

[61] R. Jayaraman, K. Salah, and N. King, "Improving opportunities in healthcare supply chain processes via the internet of things and blockchain technology," *International Journal of Healthcare Information Systems and Informatics (IJHISI),* vol. 14, no. 2, pp. 49-65, 2019.

[62] K. Peterson, R. Deeduvanu, P. Kanjamala, and K. Boles, "A blockchain-based approach to health information exchange networks," in *Proc. NIST Workshop Blockchain Healthcare*, 2016, pp. 1-10.

[63] S. Zhang and J.-H. Lee, "Analysis of the main consensus protocols of blockchain," *ICT express,* vol. 6, no. 2, pp. 93-97, 2020.

[64] M. Crosby, P. Pattanayak, S. Verma, and V. Kalyanaraman, "Blockchain technology: Beyond bitcoin," *Applied Innovation,* vol. 2, no. 6-10, p. 71, 2016.

[65] T. Moran and I. Orlov, "Proofs of Space-Time and Rational Proofs of Storage," *IACR Cryptol. ePrint Arch.,* vol. 2016, no., p. 35, 2016.

[66] A. K. Talukder, M. Chaitanya, D. Arnold, and K. Sakurai, "Proof of disease: A blockchain consensus protocol for accurate medical decisions and reducing the disease burden," in *2018 IEEE SmartWorld, ubiquitous intelligence & computing, advanced & trusted computing, scalable computing & communications, cloud & big data computing, internet of people and smart city innovation (SmartWorld/SCALCOM/UIC/ATC/CBDCom/IOP/SCI)*, 2018, pp. 257-262.

[67] J. Chen, X. Ma, M. Du, and Z. Wang, "A blockchain application for medical information sharing," in *2018 IEEE International Symposium on Innovation and Entrepreneurship (TEMS-ISIE)*, 2018, pp. 1-7.

[68] Y. Yuan and F.-Y. Wang, "Towards blockchain-based intelligent transportation systems," in *2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC)*, 2016, pp. 2663-2668.

[69] M. Singh and S. Kim, "Blockchain based intelligent vehicle data sharing framework," *arXiv preprint arXiv:1708.09721*, no., 2017.

[70] Z. Yang, K. Zheng, K. Yang, and V. C. Leung, "A blockchain-based reputation system for data credibility assessment in vehicular networks," in *2017 IEEE 28th annual international symposium on personal, indoor, and mobile radio communications (PIMRC)*, 2017, pp. 1-5.

[71] R. Singh and L. Benyoucef, "A consensus based group decision making methodology for strategic selection problems of supply chain coordination," *Engineering Applications of Artificial Intelligence,* vol. 26, no. 1, pp. 122-134, 2013.

[72] K. Leng, Y. Bi, L. Jing, H.-C. Fu, and I. Van Nieuwenhuyse, "Research on agricultural supply chain system with double chain architecture based on blockchain technology," *Future Generation Computer Systems,* vol. 86, no., pp. 641-649, 2018.

[73] J. Gubbi, R. Buyya, S. Marusic, and M. Palaniswami, "Internet of Things (IoT): A vision, architectural elements, and future directions," *Future generation computer systems,* vol. 29, no. 7, pp. 1645-1660, 2013.

[74] S. Li, G. Oikonomou, T. Tryfonas, T. M. Chen, and L. Da Xu, "A distributed consensus algorithm for decision making in service-oriented internet of things," *IEEE Transactions on Industrial Informatics,* vol. 10, no. 2, pp. 1461-1468, 2014.

[75] L. Lamport, "Paxos made simple," *ACM Sigact News,* vol. 32, no. 4, pp. 18-25, 2001.

[76] G. Brambilla, M. Amoretti, and F. Zanichelli, "Using blockchain for peer-to-peer proof-of-location. arXiv 2016," *arXiv preprint arXiv:1607.00174*, no.

[77] D. Fu and L. Fang, "Blockchain-based trusted computing in social network," in *2016 2nd IEEE International Conference on Computer and Communications (ICCC)*, 2016, pp. 19-22.
[78] M. O'Dair, "Music on the blockchain: blockchain for creative industries research cluster," *Middlesex University Report,* vol. 1, no., pp. 4-24, 2016.
[79] A. Spielman, "Blockchain: digitally rebuilding the real estate industry," Massachusetts Institute of Technology, 2016.
[80] G. Foroglou and A.-L. Tsilidou, "Further applications of the blockchain," in *12th student conference on managerial science and technology*, 2015, pp. 1-8.
[81] A. Baum, "PropTech 3.0: the future of real estate," no., 2017.
[82] Z. Guan, G. Si, X. Zhang, L. Wu, N. Guizani, X. Du*, et al.*, "Privacy-preserving and efficient aggregation based on blockchain for power grid communications in smart communities," *IEEE Communications Magazine,* vol. 56, no. 7, pp. 82-88, 2018.
[83] J. Blocki and H.-S. Zhou, "Designing proof of human-work puzzles for cryptocurrency and beyond," in *Theory of Cryptography Conference*, 2016, pp. 517-546.
[84] K. Li, H. Li, H. Hou, K. Li, and Y. Chen, "Proof of vote: A high-performance consensus protocol based on vote mechanism & consortium blockchain," in *2017 IEEE 19th International Conference on High Performance Computing and Communications; IEEE 15th International Conference on Smart City; IEEE 3rd International Conference on Data Science and Systems (HPCC/SmartCity/DSS)*, 2017, pp. 466-473.
[85] E. Buchman, "Tendermint: Byzantine fault tolerance in the age of blockchains," 2016.
[86] S. Namasudra, G. C. Deka, P. Johri, M. Hosseinpour, and A. H. Gandomi, "The revolution of blockchain: State-of-the-art and research challenges," *Archives of Computational Methods in Engineering,* vol. 28, no. 3, pp. 1497-1515, 2021.
[87] M. Castro and B. Liskov, "Practical byzantine fault tolerance," in *OSDI*, 1999, pp. 173-186.
[88] M. Castro and B. Liskov, "Practical Byzantine fault tolerance and proactive recovery," *ACM Transactions on Computer Systems (TOCS),* vol. 20, no. 4, pp. 398-461, 2002.
[89] D. Ongaro and J. Ousterhout, "The raft consensus algorithm," no., 2015.
[90] D. Huang, X. Ma, and S. Zhang, "Performance analysis of the raft consensus algorithm for private blockchains," *IEEE Transactions on Systems, Man, and Cybernetics: Systems,* vol. 50, no. 1, pp. 172-181, 2019.
[91] C. Ehmke, F. Wessling, and C. M. Friedrich, "Proof-of-property: a lightweight and scalable blockchain protocol," in *Proceedings of the 1st International Workshop on Emerging Trends in Software Engineering for Blockchain*, 2018, pp. 48-51.
[92] X. Li, P. Jiang, T. Chen, X. Luo, and Q. Wen, "A survey on the security of blockchain systems," *Future Generation Computer Systems,* vol. 107, no., pp. 841-853, 2020.
[93] M. Saad, M. T. Thai, and A. Mohaisen, "POSTER: deterring ddos attacks on blockchain-based cryptocurrencies through mempool optimization," in *Proceedings of the 2018 on Asia Conference on Computer and Communications Security*, 2018, pp. 809-811.
[94] S. Sayeed and H. Marco-Gisbert, "Assessing blockchain consensus and security mechanisms against the 51% attack," *Applied Sciences,* vol. 9, no. 9, p. 1788, 2019.
[95] C. Ye, G. Li, H. Cai, Y. Gu, and A. Fukuda, "Analysis of security in blockchain: Case study in 51%-attack detecting," in *2018 5th International Conference on Dependable Systems and Their Applications (DSA)*, 2018, pp. 15-24.
[96] K. Wüst and A. Gervais, "Ethereum eclipse attacks," ETH Zurich2016.
[97] K. Nayak, S. Kumar, A. Miller, and E. Shi, "Stubborn mining: Generalizing selfish mining and combining with an eclipse attack," in *2016 IEEE European Symposium on Security and Privacy (EuroS&P)*, 2016, pp. 305-320.

[98] R. Sahay, G. Geethakumari, and B. Mitra, "A novel blockchain based framework to secure IoT-LLNs against routing attacks," *Computing,* vol. 102, no., pp. 2445-2470, 2020.

[99] M. Saad, V. Cook, L. Nguyen, M. T. Thai, and A. Mohaisen, "Partitioning attacks on bitcoin: Colliding space, time, and logic," in *2019 IEEE 39th International Conference on Distributed Computing Systems (ICDCS)*, 2019, pp. 1175-1187.

[100] M. Pilkington, "Blockchain technology: principles and applications," in *Research handbook on digital transformations*, ed: Edward Elgar Publishing, 2016.

[101] H. Lee, M. Shin, K. S. Kim, Y. Kang, and J. Kim, "Recipient-oriented transaction for preventing double spending attacks in private blockchain," in *2018 15th Annual IEEE International Conference on Sensing, Communication, and Networking (SECON)*, 2018, pp. 1-2.

[102] A. Malik, S. Gautam, S. Abidin, and B. Bhushan, "Blockchain technology-future of IoT: including structure, limitations and various possible attacks," in *2019 2nd International Conference on Intelligent Computing, Instrumentation and Control Technologies (ICICICT)*, 2019, pp. 1100-1104.

[103] A. Alkhalifah, A. Ng, A. Kayes, J. Chowdhury, M. Alazab, and P. A. Watters, "A taxonomy of blockchain threats and vulnerabilities," in *Blockchain for Cybersecurity and Privacy*, ed: CRC Press, 2020, pp. 3-28.

[104] D. Dasgupta, J. M. Shrein, and K. D. Gupta, "A survey of blockchain from security perspective," *Journal of Banking and Financial Technology,* vol. 3, no. 1, pp. 1-17, 2019.

[105] M. Alizadeh, K. Andersson, and O. Schelén, "A Survey of Secure Internet of Things in Relation to Blockchain," *Journal of Internet Services and Information Security,* vol. 3, no., 2020.

[106] M. Saad, J. Spaulding, L. Njilla, C. Kamhoua, S. Shetty, D. Nyang*, et al.*, "Exploring the attack surface of blockchain: A systematic overview," *arXiv preprint arXiv:1904.03487*, no., 2019.

[107] R. C. Lunardi, R. A. Michelin, C. V. Neu, H. C. Nunes, A. F. Zorzo, and S. S. Kanhere, "Impact of consensus on appendable-block blockchain for IoT," in *Proceedings of the 16th EAI International Conference on Mobile and Ubiquitous Systems: Computing, Networking and Services*, 2019, pp. 228-237.

[108] T. McGhin, K.-K. R. Choo, C. Z. Liu, and D. He, "Blockchain in healthcare applications: Research challenges and opportunities," *Journal of Network and Computer Applications,* vol. 135, no., pp. 62-75, 2019.

[109] M. Li, L. Shen, and G. Q. Huang, "Blockchain-enabled workflow operating system for logistics resources sharing in E-commerce logistics real estate service," *Computers & Industrial Engineering,* vol. 135, no., pp. 950-969, 2019.

[110] M. Csernai, A. Gulyas, Z. Heszberger, S. Molnar, and B. Sonkoly, "Congestion control and network management in Future Internet," *Infocommunications Journal*, no., p. 14, 2009.

[111] M. Gerlach and F. Guttler, "Privacy in vanets using changing pseudonyms-ideal and real," in *2007 IEEE 65th Vehicular Technology Conference-VTC2007-Spring*, 2007, pp. 2521-2525.

[112] A. M. Carianha, L. P. Barreto, and G. Lima, "Improving location privacy in mix-zones for VANETs," in *30th IEEE International Performance Computing and Communications Conference*, 2011, pp. 1-6.

[113] M. Raya and J.-P. Hubaux, "Securing vehicular ad hoc networks," *Journal of computer security,* vol. 15, no. 1, pp. 39-68, 2007.

[114] D. Förster, F. Kargl, and H. Löhr, "PUCA: A pseudonym scheme with strong privacy guarantees for vehicular ad-hoc networks," *Ad Hoc Networks,* vol. 37, no., pp. 122-132, 2016.

[115] F. Schaub, Z. Ma, and F. Kargl, "Privacy requirements in vehicular communication systems," in *2009 International Conference on Computational Science and Engineering*, 2009, pp. 139-145.

[116] J. Wang, Y. Zhang, Y. Wang, and X. Gu, "RPRep: A robust and privacy-preserving reputation management scheme for pseudonym-enabled VANETs," *International Journal of Distributed Sensor Networks,* vol. 12, no. 3, p. 6138251, 2016.

[117] U. Rajput, F. Abbas, and H. Oh, "A hierarchical privacy preserving pseudonymous authentication protocol for VANET," *IEEE Access,* vol. 4, no., pp. 7770-7784, 2016.

[118] L. Zhang, Q. Wu, J. Domingo-Ferrer, B. Qin, and C. Hu, "Distributed aggregate privacy-preserving authentication in VANETs," *IEEE Transactions on Intelligent Transportation Systems,* vol. 18, no. 3, pp. 516-526, 2016.

[119] H. Li, L. Pei, D. Liao, G. Sun, and D. Xu, "Blockchain meets VANET: An architecture for identity and location privacy protection in VANET," *Peer-to-Peer Networking and Applications,* vol. 12, no. 5, pp. 1178-1193, 2019.

[120] M. Wagner and B. McMillin, "Cyber-physical transactions: A method for securing VANETs with blockchains," in *2018 IEEE 23rd Pacific Rim International Symposium on Dependable Computing (PRDC)*, 2018, pp. 64-73.

[121] T. Salman, M. Zolanvari, A. Erbad, R. Jain, and M. Samaka, "Security services using blockchains: A state of the art survey," *IEEE Communications Surveys & Tutorials,* vol. 21, no. 1, pp. 858-880, 2018.

[122] Z. Lu, Q. Wang, G. Qu, and Z. Liu, "Bars: a blockchain-based anonymous reputation system for trust management in vanets," in *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)*, 2018, pp. 98-103.

[123] A. Lei, Y. Cao, S. Bao, D. Li, P. Asuquo, H. Cruickshank*, et al.*, "A blockchain based certificate revocation scheme for vehicular communication systems," *Future Generation Computer Systems,* vol. 110, no., pp. 892-903, 2020.

[124] D. Zheng, C. Jing, R. Guo, S. Gao, and L. Wang, "A traceable blockchain-based access authentication system with privacy preservation in VANETs," *IEEE Access,* vol. 7, no., pp. 117716-117726, 2019.

[125] L. Benarous, B. Kadri, and A. Bouridane, "Blockchain-Based Privacy-Aware Pseudonym Management Framework for Vehicular Networks," *Arabian Journal for Science and Engineering*, no., pp. 1-17, 2020.

[126] K. Shi, L. Zhu, C. Zhang, L. Xu, and F. Gao, "Blockchain-based multimedia sharing in vehicular social networks with privacy protection," *Multimedia Tools and Applications*, no., pp. 1-21, 2020.

[127] J. Cui, F. Ouyang, Z. Ying, L. Wei, and H. Zhong, "Secure and Efficient22Data Sharing Among Vehicles Based on Consortium Blockchain," IEEE23Transactions on Intelligent Transportation Systems, no., 2021.

[128] J. Ma, T. Li, J. Cui, Z. Ying, and J. Cheng, "Attribute-Based Secure25Announcement Sharing among Vehicles Using Blockchain," IEEE Internet26of Things Journal, no., 2021

[129] S. Bao, Y. Cao, A. Lei, P. Asuquo, H. Cruickshank, Z. Sun*, et al.*, "Pseudonym Management Through Blockchain: Cost-Efficient Privacy Preservation on Intelligent Transportation Systems," *IEEE Access,* vol. 7, no., pp. 80390-80403, 2019.

[130] L.-N. Lundbæk, D. Janes Beutel, M. Huth, S. Jackson, L. Kirk, and R. Steiner, "Proof of Kernel Work: a democratic low-energy consensus for distributed access-control protocols," *Royal Society open science,* vol. 5, no. 8, p. 180422, 2018.

[131] S. Bao, W. Hathal, H. Cruickshank, Z. Sun, P. Asuquo, and A. Lei, "A lightweight authentication and privacy-preserving scheme for VANETs using TESLA and Bloom Filters," *ICT Express,* vol. 4, no. 4, pp. 221-227, 2018.

[132] J. B. Kenney, "Dedicated short-range communications (DSRC) standards in the United States," *Proceedings of the IEEE,* vol. 99, no. 7, pp. 1162-1182, 2011.

[133] C. Dannen, *Introducing Ethereum and solidity* vol. 318: Springer, 2017.

[134] I. Grigg, "Eos-an introduction," *White paper. https://whitepaperdatabase. com/eos-whitepaper*, no., 2017.

[135] Y. Lewenberg, Y. Bachrach, Y. Sompolinsky, A. Zohar, and J. S. Rosenschein, "Bitcoin mining pools: A cooperative game theoretic analysis," in *Proceedings of the 2015 International Conference on Autonomous Agents and Multiagent Systems*, 2015, pp. 919-927.

[136] C. Cachin, "Architecture of the hyperledger blockchain fabric," in *Workshop on distributed cryptocurrencies and consensus ledgers*, 2016.

[137] V. Dhillon, D. Metcalf, and M. Hooper, "The hyperledger project," in *Blockchain enabled applications*, ed: Springer, 2017, pp. 139-149.

[138] M. Valenta and P. Sandner, "Comparison of ethereum, hyperledger fabric and corda," *Frankfurt School Blockchain Center,* vol. 8, no., 2017.

[139] R. G. Brown, J. Carlyle, I. Grigg, and M. Hearn, "Corda: an introduction," *R3 CEV, August,* vol. 1, no., p. 15, 2016.

[140] T. T. A. Dinh, R. Liu, M. Zhang, G. Chen, B. C. Ooi, and J. Wang, "Untangling blockchain: A data processing view of blockchain systems," *IEEE Transactions on Knowledge and Data Engineering,* vol. 30, no. 7, pp. 1366-1385, 2018.