Investigation of Blockchain-based emerging solutions for various centralized applications

Amruranshu Panigrahi

Department of Computer Science and Engineering

Siksha O Anusandhan (Deemed to be University)

A thesis submitted for the degree of

Doctor of Philosophy

date

SIKSHA 'O' ANUSANDHAN

(A Deemed to be University declared u/s 3 of UGC Act, 1956)
Accredited (3rd Cycle) by NAAC with A++ Grade

Notification No.SOA/CoE/Ph.D/1077/Ph387/2023 **Date: 09.11.2023**

In pursuance of the approval accorded by the Vice-Chancellor under Rule17.4 of the regulations, 2012, governing Ph.D. programme in Engineering & Technology, **Amrutanshu Panigrahi, Regn.No-2081001006** is provisionally declared to be awarded the degree of Doctor of Philosophy in **Computer Science and Engineering** having his research in the subject, under the title **"Investigation of Blockchain-based emerging solutions for various centralized applications".**

Controller of Examinations

Memo. No.SOA/CoE/Ph.D/1077/Ph388 /2023 **Date: 09.11.2023**

Copy to:

1. Mr. Amrutanshu Panigrahi, At/Po-Sirapur, Via-C.B.Pur, Dist-Balasore-756055, Odisha, Email: amrutansup89@gmail.com, Mob-7070784109 for information.
2. Prof. Amlan Chakrabarti, Professor & Director, School of IT, University of Calcutta, JD-2, JD Block, Sector-III, Bidhan Nagar, Kolkata-700106, West Bengal, Email: acakcs@caluniv.ac.in, Mob-9831129520, Indian examiner for information.
3. Dr. Rourab Paul, Associate Professor, Department of Computer Science & Engineering, Faculty of Engineering & Technology (ITER), Siksha 'O' Anusandhan Deemed to be University, Jagamohan Nagar, Khandagiri Square, Bhubaneswar-751030, Email: rourabpaul@soa.ac.in, Mob: 9836057376, Supervisor & DAC Member for information.
4. Prof. (Dr.) Ajit Kumar Nayak, Professor & HOD, Department of Computer Science and Information Technology, Faculty of Engineering & Technology (ITER), Siksha 'O' Anusandhan Deemed to be University, Bhubaneswar-751030, Email: ajitnayak@soa.ac.in, Mob-9338749992, Co-Supervisor & DAC Member for information.
5. Prof. (Dr.) Srikanta Patnaik, Director, Interscience Institute of Management & Technology, Kantabada, Bhubaneswar-752054, Odisha, Email: patnaik_srikanta@yahoo.co.in, Mob-9937167777, DAC Member for information.
6. Prof.(Dr.) Benudhar Sahu, Professor, Department of ECE, Faculty of Engineering & Technology (ITER), Siksha 'O' Anusandhan Deemed to be University, Bhubaneswar-751030, Email: benudharsahu@soa.ac.in, Mob-9437883473, DAC Member for information.
7. Dr. Manoranjan Parhi, Associate Professor, Centre for Data Science, Faculty of Engineering & Technology (ITER), Siksha 'O' Anusandhan Deemed to be University, Bhubaneswar-751030, Email: manoranjanparhi@soa.ac.in, DAC Member for information.
8. The HOD, Department of Computer Science and Engineering, Faculty of Engineering & Technology (ITER), Siksha 'O' Anusandhan Deemed to be University, Bhubaneswar-751030, Email: hod.cse.iter@soa.ac.in, Ex-officio Chairman for information.
9. The Dean, Faculty of Engineering & Technology (ITER) Siksha 'O' Anusandhan Deemed to be University, Bhubaneswar-30 for information.
10. The Dean, Research, Siksha 'O' Anusandhan Deemed to be University, Bhubaneswar for information.
11. The IQAC (iqacell@soa.ac.in), Siksha 'O' Anusandhan Deemed to be University, Bhubaneswar for information.
12. The Chief Librarian, Siksha 'O' Anusandhan Deemed to be University, Bhubaneswar-751030, Email: dolababuramesh@soa.ac.in for information.
13. The Registrar, Siksha 'O' Anusandhan Deemed to be University, Bhubaneswar for information.
14. All Pro-Vice Chancellor, Siksha 'O' Anusandhan Deemed to be University, Bhubaneswar for information.
15. The Secretary to the Vice-Chancellor, Siksha 'O' Anusandhan Deemed to be University, Bhubaneswar for kind information of the Vice-Chancellor.
16. Dy. Controller of Examinations, Siksha 'O' Anusandhan Deemed to be University, Bhubaneswar for information & necessary action
17. The Director of Admissions, Siksha 'O' Anusandhan Deemed to be University, Bhubaneswar for information.
18. The Chairman, UGC, Bahadurshah Zafar Marg, New Delhi-110002, Email: cm.ugc@nic.in for information.
19. The Director, Information & Library Network (INFLIBNET) Centre, Infocity, Gandhinagar-382007, Gujarat, Email: director@inflibnet.ac.in for information.
20. The Library and Documentation Division, Association of Indian Universities(AIU),AIU House, 16-Comrade Indrajit Gupta (Kotla) Marg, New Delhi-110002,Email: library@aiu.ac.in for information.
21. The Secretary General, Association of Indian Universities, AIU House,16-Comrade Indrajit Gupta Marg (Kotla Marg) New Delhi-110002,Email: info@aiu.ac.in for information.
22. The Editor, University News, Association of Indian Universities, AIU House,16-Comrade Indrajit Gupta Marg (Kotla Marg), New Delhi-110002,Email: universitynews@aiu.ac.in for information.
23. Guard File.

Controller of Examinations

I would like to dedicate this thesis to my loving parents, wife, daughter, sisters, brothers, and friends ...

Acknowledgements

Without the individuals whose unwavering support and encouragement made it possible, it would be impossible to feel the full sense of accomplishment that comes from completing a difficult endeavor.

First of all, I want to express my deep sense of gratitude to my Supervisor, Dr.Rourab Paul, & Co-supervisor Prof. Ajit Kumar Nayak, for all of their kind help and encouragement during my Ph.D. work. The guidance they provided was valuable to me as I worked on my thesis's research and writing. Their expertise and enthusiasm for the research inspire me to keep going even when the going gets tough, and this has helped me to be developed into a more resilient and successful researcher.

I would like to extend my gratitude to the esteemed Vice Chancellor, Registrar, and Controller of Examination, Siksha 'O' Anusandhan (Deemed to be University) Bhubaneswar, without whom it would not have been possible to finish the thesis on time. In addition, I'd like to express my gratitude to Prof. (Dr.) P. K. Sahoo, Dean, Institute of Technical Education and Research (ITER), Siksha 'O' Anusandhan (Deemed to be University) Bhubaneswar, for his assistance with the infrastructure and resources I needed during my study.

For all the help and consideration she has given me as a scholar, I would also like to thank Prof. (Dr.) Debahuti Mishra, Head of the Department, Computer Science & Engineering, Institute of Technical Education & Research (ITER), Siksha 'O' Anusandhan (Deemed to be University).

I would like to express my deepest gratitude to Prof. (Dr.) J. K. Nath, Dean of Research & Development, Institute of Technical Education and Research (ITER), Siksha 'O' Anusandhan (Deemed to be University), for his constant enthusiasm and insightful feedback. I would also like to extend my heartful thanks to my Doctoral Advisory Committee members, Prof. Srikanta Patnaik, Prof. Benudhar Sahu, and Dr. Manoranjan Parhi, for their insightful feedback and guidance.

For their continuous support, patience, and helpfulness throughout my Ph.D. work, I'd like to express my heartfelt thanks to my parents, Padma Lochan Panigrahi and Damayanti Panigrahi; wife, Swagatika Panda; my adorable

daughter, Loukya Panigrahi; sisters, Snigdha Panigrahi, Snirupa Panigrai; brothers-in-law, Dr. Manorajan Kar, Dr. Jyoti Shankar Mishra; uncle Purna Chandra Pati, aunty Shantilata Pati, and brother Ptitam Kumar Panda. In addition, I would also like to extend my thank to my father-in-Law Gobinda Chandra Panda, and mother-in-Law Kalyani Panda, for their kindness and generosity.

I'd want to thank everyone who encouraged me and helped me think critically while I was doing this research. Especially I would like thank to Mr. Abhilash Pati, Mr. Bibhuprasad Sahu, and many others for their unwavering support and helpfulness. My memories of our time together will last forever.

Finally, I want to thank the almighty Lord Shree Jagannatha for being the impetus behind the completion of my Ph.D. work.

Abstract

Since the emergence of blockchain technology in the form of Bitcoin by Satoshi Nakamoto, its development has progressed rapidly that has attracted the attention of various researchers in academia and industry. Blockchain technology is becoming an increasingly secure and effective way to share information in various industries, including finance, healthcare, supply chain management (SCM), and the Internet of Things (IoT). Blockchain is a decentralized system implemented in a peer-to-peer network which can store transactional records in a distributed database known as a distributed ledger. Every active peer or node in such a network has a copy of the distributed ledger. Decentralization, immutability, security, and transparency make Blockchain technology one of the most promising and prominent technology in internet-based communication.

Most of the real-life applications deal with enormous amounts of data exchange administrated by authorized access of users or nodes. The IoT with 5G communication increases the volume of data and makes the problem of authorization of data access more complex. The fundamental requirement of data exchange policies among multiple nodes also demand immutability and decentralization to avoid single-point failure. Most of the conventional applications like Electronic Health Record (EHR), Public Key Infrastructure (PKI), Supply Chain Management, IoT applications, e-governance and banking suffer from single-point failure, security risks, limited scalability, and lack of transparency. In this regard, the adoption of blockchain in EHR, PKI and other data exchange applications can address all the above-mentioned issues.

For example, a simple blockchain-based decentralized application (dApp) for healthcare data can eliminate all the potential barriers of a centralized application. In health care system (HCS) application decentralization and security are the two most essential features required to exchange data between patients and doctors. The absence of the above-mentioned features in HCS can cause difficulty for patients in accessing their health data. Similarly, in conventional PKI, a centralized third-party validator or CA for generating, issuing, and managing digital certificates may cause a single point of failure. Communications established by conventional PKIs rely on third-party centralized CAs,

which fundamentally breaches the security paradigm. As a result, the literature has reported many incidents of malicious CAs. Additionally, conventional PKI does not provide an effective way to detect the malicious behavior of the validator or CA. These limitations of conventional PKI make it challenging to be used as a solution for secure data exchange.

The adoption of blockchain technology in HCS makes it more productive and effective. The advantages of blockchain technology such as decentralization and immutability can facilitate better and secured management of electronic health records (EHR), electronic medical records (EMR) for various medical devices, billing, and telemedicine systems. The selection of different CAs for different transactions can eliminate the single-point failure limitation of conventional PKI. Additionally, the use of DLT can identify the malicious behavior of the selected CA. Efficient smart contract can prevent many attacks on BC-PKI, and the adoption of efficient consensus algorithms along with clustering to reduce searching space of CA selection process can also reduce the significant amount of computational overhead of BC-PKI. Moreover, the consensus algorithm solves the issue of the third-party CA.

The objective of the thesis is divided into three parts: The first part develops a basic blockchain-based dApp for EHR. The second part proposes and implements a new light weight smart contract which addresses different attacks on blockchain-based PKI while the third part uses cluster algorithms along with a modified consensus algorithm to reduce the computational overhead of BC-PKIs.

In the first part, a basic dApp for EHR was developed to store medical data and facilitate its exchange between patients and doctors. The developed dApp allows patients to choose their preferred doctor. The selected doctor can access the EHR of the patient and diagnose it. Once the patient is diagnosed, then she/he has to make the payment to that corresponding doctor. For this purpose, each patient is attached to MetaMask digital wallet through their private key. MetaMask helps the patient to transfer the fee in terms of Ethereum coin named as ETH to the doctor using doctor's public key. The developed dApp provides a simple blockchain-based communication between the patient and doctor. However, the security concern in terms of different attacks has not been addressed in this work.

The BC-PKI is reported as a second part of the thesis which addresses different attacks such as Denial of Service (DoS), Man in the Middle Attack (MITM), Distributed Denial of Service (DDoS), 51%, Injection attacks, Routing attacks, and Eclipse attack. This BC-PKI uses a lightweight smart contract to validate

the identity of the node and the CA. The smart contract used in BC-PKI sets a maximum threshold for all nodes. A counter in each node counts the number of times that node becomes a CA. If the counter in each node exceeds the threshold value, the proposed smart contract will not allow that node as a CA. The Delegated Proof of Stake (DPoS) consensus algorithm used in this model makes it suitable for lightweight applications. Undoubtedly, the developed BC-PKI provides a secure framework for internet-based communication. However, this BC-PKI does not aim to reduce the computational overhead caused by the CA selection process.

The decentralization feature of public and private blockchain-based applications is achieved by selecting different nodes as validators or Certificate Authority (CA) for each transaction. The network may have a large number of transactions and participants. For a large network, selecting a CA for each transaction may cause a significant amount of block propagation delay, which can reduce network efficiency drastically. The third part of the thesis proposes a different approach to increase the efficiency of BC-PKI. The developed approach creates clusters of participant nodes based on their validation time, response time, and trust. This method selects a cluster based on the budget of response time and validation time given by the node that intends to start a transaction. Thereafter, the node which has the highest trust in that cluster is chosen as a CA for the next transaction. Instead of searching on all participant nodes, our approach searches on the nodes of the chosen cluster which reduces the searching space of the CA selection process. This research work adopts a trust evaluation approach where the trust factor is quantified based on its experience and reputation. The node trust is re-evaluated after every successful and unsuccessful transaction. A node that performs more successful transactions has more trust value. The node that has a higher trust value has a higher probability to be selected as a CA for a transaction.

The works as presented in this thesis use the open-source development environment Ethereum platform ($GETH$) with 100 nodes. Additionally, the $Truffle\ Suite$ was used to deploy and test the smart contract on the blockchain. The $Remix\ IDE$ with $Solidity\ v0.4.24$ scripting language was used to develop the smart contract. For the dApp, the $MetaMask$ wallet was used to facilitate payments between doctors and patients. To develop the dApp and PKI, a Windows 10 OS, 8GB RAM, Intel i5 processor with a clock speed of 2.8 GHz, 1TB HDD, and 500GB SSD is used.

The author firmly believes the research work should be open sourced so that the research community can reap maximum benefits. Following this motiva-

tion, the author has uploaded all the relevant codes pertaining to this thesis work at **....git hub link...**

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Before the advent of Blockchain technology, there was no decentralized mechanism to govern online actions which can ensure non-repudiation of data. The two parties or nodes involved in a transaction do not have faith in the other's commitment to keep their information unaltered for commercial interest. The modification of information is allowed only if both the parties have the relevant knowledge or provide their consent for the same. Different group of individuals nodes do not rely on any kind of transaction or communication without the presence of central authority. This issue commonly refer to as "*Byzantine Generals Dilemma*" which states the hardness of agreeing on anything without relying on central authorities. In this classic problem, the Byzantine army is divided into multiple battalions where each battalion reports to a different generals. Generals of all battalions communicate through a common messenger. In response to the message, they agree on a strategy which calls for simultaneous attacks from all battalions. This strategy is likely to be sabotaged by traitors who intercept or alter their communications. The main challenge of all generals is to agree on a common contract with the presence of imposters.

The issue of Byzantine Generals problem may be solved using Blockchain technology by adopting a probabilistic approach. The movement and distributed storage of data over a network of computer nodes not only leads to an improvement in transparency but dependability as well. As a result, the probability of attackers to compromise the integrity of a distributed database considerably decreases. Attacks on this network may be possible only when attackers have significant amount of network processing power. Moreover, the protocols of Blockchain have the ability to guarantee the integrity of the transactions.

Blockchain is a chronological archive of transactions. These transactions are organized into blocks which are validated by other computer nodes connected with the Blockchain network. The data in each block is validated by an intricate mathematical equation or a set of hash functions, which preserves the data integrity as well. The Blockchain database is replicated on all the nodes which are part of the network. These nodes are regularly

synced to access the most latest and accurate version of the shared database.

Blockchain technology (BCT) is a framework for performing and recording transactions in a peer-to-peer network. This storage of validated transaction is called a distributed ledger. Instead of validation of a central node, Blockchain technology is rather based on multiple nodes[1, 2]. Initially, in 2009 Blockchain technology has been introduced in terms of Bitcoin as a form of cryptocurrency. Since Blockchain came into the picture, this technology has attracted a lot of attention across a wide range of industries. It is becoming an emerging solution in transforming the traditional communication system into a distributed one by implementing a distributed ledger (DLT) which is made available to every participant in the peer-to-peer network. The fundamental idea behind the DLT is to store all the blocks consisting of transaction data. Each block in DLT includes two things: block header and transactional data. The block header consists of the hash of the previous block, timestamp, and transaction information. The Merkel root tree used in Blockchain is a data structure where the leaf node represents the cryptographic hash of data and the non-leaf node presents the hash combination of its corresponding child nodes. Figure 1.1 represents the block structure of a block in DLT.

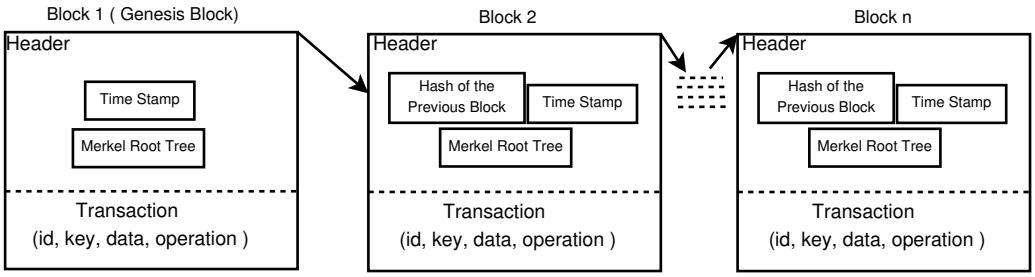

Figure 1.1: Block Structure in Blockchain Technology

The use of decentralization features simply removes the potential barrier of a central authority which makes the network more consistent and secure during communication. The use of diverse keys and hash functions in a Blockchain network increases the security of communication in the network [4]. Distributed ledger technology (DLT) is a distributed database that contains transaction information and participant information that is made accessible to all participating nodes while maintaining data integrity [5,6]. Consensus and the Smart contract are two fundamental elements of Blockchain which make the network fault-tolerant and resilient [7]. There are three forms of Blockchain technology : private Blockchain, known as the permission-oriented Blockchain, public Blockchain known as the permissionless Blockchain, and hybrid Blockchain, which combines the features of both private and public Blockchain [8,9,10]. Blockchain is still in its early stages. However, it has already elicited strong positive responses and excitement from academia and variety of industries, like banking, IoT, healthcare system etc. The medical business may be benefited from the use of Blockchain and distributed ledger technology (DLT) [11,12].

The advantages of Blockchain technology, such as decentralization, security, protection, as well as bitcoin and smart contracts based on cryptographic computations [38], have huge potential to overcome the present problems associated with the centralized communication system.

1.1 Blockchain Evolution

When it comes to Blockchain technology, there are four distinct versions accessible, ranging from Blockchain 1.0 to Blockchain 4.0 [39]. The various evolution of the Blockchain is reflected in Figure 1.2.

- **Blockchain 1.0:** It dealt with cryptocurrencies such as bitcoin, and it allowed for safe transactions to be carried out between various users. This also makes it possible to use in DLT. Bitcoin mining scams, exchange scams, and wallet frauds are just a few of the well-known drawbacks of digital currency production.

- **Blockchain 2.0:** This version of the Blockchain has the smart contract as a key concept. Smart contracts are a collection of predefined codes which is triggered during a transaction. when the transaction reaches a certain condition, hence reducing the transaction's validation costs. However, the fundamental botteleneck is that once a smart contract is in perform, it is very difficult to stop it from continuing to operate.

- **Blockchain 3.0:** The major emphasis is on the scalable user interface for the user end, as well as the decentralized application. Digital application programming interfaces (DAPPS) are made up of a user-defined front end and smart contracts. Because it is dependent on a third-party API to function, the updating of DAPPS was the most significant difficulty that arose in this approach.

- **Blockchain 4.0:** It is entirely devoted to applications based on the Industry 4.0 paradigm. The smart contract, security, and a decentralized storage place are the three most important prerequisites for this plan to be successful.

1.2 Types of Blockchain

Blockchain technologies is basically classified into three different categories such as public, private, private, and consortium Blockchain.

- **Public Blockchain:** A public Blockchain network also known as a permissionless Blockchain network allows everyone to join without imposing any limitations. The

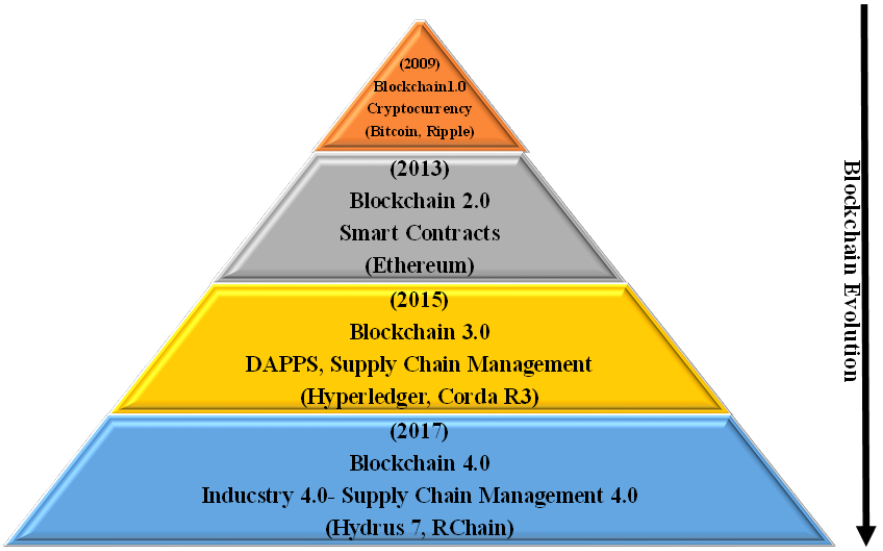

Figure 1.2: Evolution of Blockchain Technology

vast majority of cryptocurrencies are designed to operate on a public Blockchain that is regulated by a consensus method.

- **Private Blockchain:** A private Blockchain, also known as a permissioned Blockchain, gives companies the ability to limit who may access the data stored on the Blockchain. Oracle Blockchain Platform is a Blockchain that requires authorization to access.

- **Consortium Blockchain:** Consortium or Federated Blockchain combines both public and private Blockchains into a single system. As an added bonus, it allows for a predetermined approved node to be selected. In addition, B2B collaboration is commonplace. Consortium Blockchains include Hyperledger and R3CEV.

1.3 Prilimnary Study

In this section, various preliminary things such as Blockchain features, working principle of Blockchain, various Blockchain platforms, smart contracts, consensus mechanisms, public key infrastructure (PKI), and decentralized applications (DApps) are discussed.

1.3.1 Blockchain Feature

The features of Blockchain include immutability, decentralization, security, and distributed ledger technology (DLT) [**?**], [16]. Blockchain is becoming very powerful and useful due to its pervasive features.

- **Immutability:** This characteristic ensures integrity of data or transaction of nodes connected in the peer-to-peer network. The immutability property of Blockchain

allows all participating nodes to keep the copy of the transactions and it also ensures that the data can not be altered without permission of the nodes. Immutability implies that once the transaction record is updated, it is not possible to roll back any modifications by any nodes.

- **Consensus:** Each Blockchain has a consensus mechanism that assists the network in selecting transaction validators neutrally. The consensus algorithms allow a collection of participant nodes of a netowrk to decide a conclusion fast, efficiently and effectively. Even if the nodes in the network do not trust each other, they may rely on this decision-making algorithm, which is the heart of the network.There is a wide variety of consensus algorithms available, each with its own advantages and disadvantages. A consensus algorithm is essential for the operation of any Blockchain.

- **Decentralization:** The decentralization feature of the Blockchain technology eliminates the centralized authority.

- **Security:** Along with the immutability and decentralization the Blockchain enforces the security parameters such as private key, public key, and hash function to make Blockchain-based communication more dependable.

- **Distributed Ledger Technology (DLT):**DLT is a distributed database for storing the blocks and the information of all available peers in the network. The main objetive of DLT is to provide access to every nodes present in the network to maintain data integrity.

1.3.2 Working Principle of Blockchain

For each transaction, there is an equal chance for every node present in the network to become the validator. The validator for each transaction can be chosen in two ways: trustworthy selection or NONCE creation. In the trustworthy selection process, a node can only select a non-malicious validator node in which it has faith. In this case, the selection process takes less time because the initiator strictly avoids the election process and directly selects a validator for the corresponding transaction. Therfore, it reduces the computational complexity.

In this process multiple nodes may rely on a single node to validate its transactions. This situation increases the probability of becoming the certificate and key pair validator for the aformentioned single node for multiple transactions. As a result, the overall performance of the corresponding node decreases, which may have a significant impact on the overall network performance. This disadvantages of trustworthy selection process makes NONCE process more popular.

In *NONCE* or *Number only ONCe* method, the node who wants to begin a transaction broadcasts a numerical puzzle into the network. The node that solves the NONCE forward the solution to the initiator node along with the timestamp indicating when the solution was obtained. The initiator node determines the validator responsible for validating the key pair and certificate needed for the communication based on the timestamp. Finding NONCE is a very complex computation which requires dedicated computational facility or hardware. Therefore, the probability of solving the NONCE puzzle for lightweight devices like mobile phones and tablets is quite low. Hence, this method for selecting the validator fails to provide a fair chance to every availabe nodes present in the network. Figure 1.3 shows the transaction processing procedure in a Blockchain network.

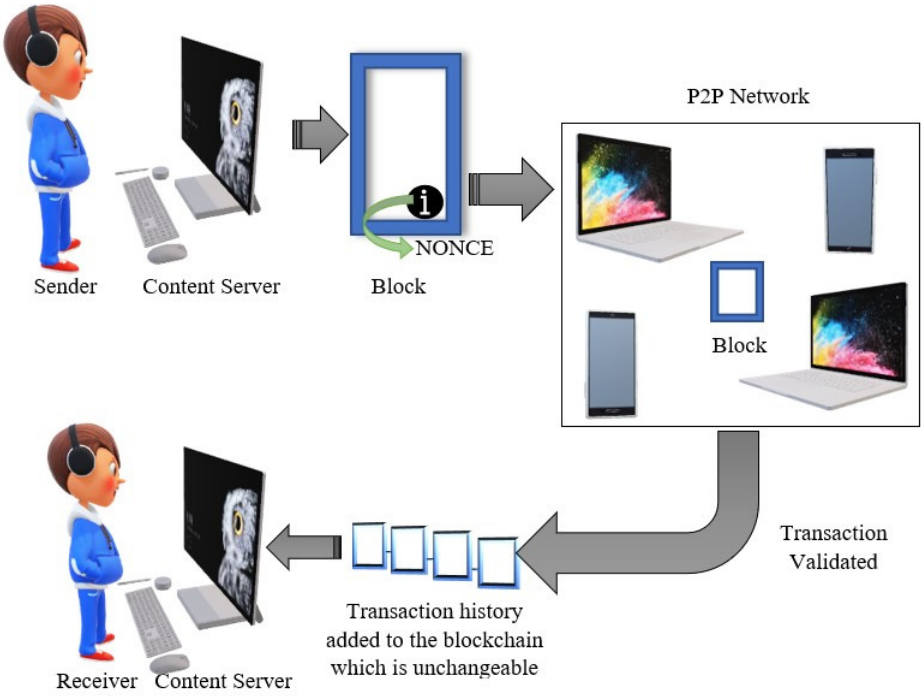

Figure 1.3: Transaction processing in Blockchain

1.3.3 Blockchain Platforms

There are three fundamental issues to consider when developing Blockchain-based applications:

- Development Platfrom of Blockchain.
- Pros and cons of Blockchain platforms.
- Consensus algorithms suitable for application.

In order to address the aforementioned issues, this section presents an overview of different Blockchain platforms that are currently available.

(i) **Bitcoin:** Bitcoin has been developed in 2008 by Satoshi Nakamoto. The main objective of this platform is to introduce the decentralization concept of financial system. There is no need for a common server in a Blockchain network to validate every transactions. For each transaction, a validator named as $Certificate\ Authority$ (CA) needs to be selected for validation process. Proof of Work (PoW) is the only consensus mechanism supported by this platform to select the CA.

(ii) **Ethereum:** After the introduction of Bitcoin technology, Ethereum emerged in 2015 as another prominent Blockchain platform. Ethereum is basically a state-based transition machine that starts with a *Genesis* block and incrementally executes different transactions. After every transaction, the information are added to the existing block continuously. The transactions in Blockchain are signed data packets which are communicated from one node to another node within the peer-to-peer network. The Ethereum has 3 main building blocks.

- **Gas Price:** It is the cost of all computations required to execute a transaction or contract. It is expressed as a number of $gWei$ which is a small fraction of Ethereum currency *ether* (ETH).
- **Gas Limit:** It is the specific amount of gas associated with a node for executing a transaction. The default gas limit present in Ethereum is 21000 units.
- **ETH:** Ether or ETH is the native cryptocurrency for the Ethereum Blockchain platform which is used to pay for any kind of activity or transaction.

Achieving overall system stability in the presence of multiple erroneous processes is a fundamental challenge in distributed computing. This typically requires the processes involved to reach a consensus on a specific data value for a transaction. The Consensus algorithm is used in these kinds of procedures. Proof of Work (PoW), Proof of Stake (PoS), and Proof of Authority (PoA) are the three most popular consensus algorithms in the Ethereum platform.

(iii) **Hyperledger:** The Linux Foundation launched the Hyperledger project as a consortium platform in December 2015. The Hyperledger has five distinct variants. Table 1.1 provides an overall comparison among different variants of this framework.

- **Burrow:** Hyperledger Burrow is developed as a permissioned Ethereum Blockchain to execute the Ethereum Virtual Machine (EVM) based smart contract in a permissioned environment. The consensus engine, the EVM, and a Remote

Procedure Call (RPC) gateway are the three key components of the node. The Byzantine Fault Tolerance (BFT) and Proof of Elapsed Time (PoET) consensus methods are used in this platform. The adoption of this in the Blockchain environment causes many significant drawbacks. One of the most significant drawbacks is the possibility that the entire network would be rendered inoperable when one-third of the trusted validators are rendered inactive.

- **Fabric:** Fabric networks are permission, which means the identities of every node that participates in the network can be verified and checked. This feature is especially helpful in sectors such as healthcare, supply chain management, finance, and insurance, where data must not be made available to unknown parties. Fabric networks are made up of channels, which are private "subnet" that enables the network to provide secure and confidential communication between two or more particular nodes. To start a transaction using the subnet, the concerned nodes must be verified by CA. In addition, all active nodes must approve the transaction to begin. This offers an extra layer of access control and is particularly helpful when all participating nodes wish to restrict exposure of the data. In addition to that, the Fabric provides a feature set known as Private Data Collection. With this set, access to certain transactions that take place on a channel may be restricted to a subset of the participants. Proof of Stake (PoS), and Delegated PoS (DPoS) are the two most used consensus mechanisms in Hyperledger Fabric.
- **Indy:**To ensure that digital identities based on Blockchains can be used in a variety of contexts, Hyperledger Indy offers a set of tools, frameworks, and reusable components for this purpose. A human-memorable name may be given to the identity once it has been generated. For use in the distributed ledger, identity is mapped to a digital identification number named as decentralized identifiers or DID. These DIDs can be used for further communications.
- **Iroha:** Hyperledger Iroha is a Blockchain platform for developing secure, reliable, and quickly deployable applications. It is a permissioned Blockchain that uses the $CrashFault$ consensus methodology to work efficiently in the presence of faulty and damaged systems. In addition, the Iroha also used the $Yet - Another - Consensus(YAC)$ mechanism to perform the transaction. It is mainly used to develop Blockchain-based application for various industry that requires DLT implementation. Multisignature is a key feature of Iroha. Unlike other Blockchain platforms, $Iroha$ requires multiple signatures from different nodes to validate the transaction. This platform is mainly used in the financial sector to develop the KYC system.
- **Sawtooth:** Hyperledger Sawtooth is an enterprise-based Blockchain solution

to develop and deploy distributed ledgers. It keeps the system secure while making it easier to build applications since it separates the core ledger system from the environment in which those applications run. Using this platform the developer can develop different applications which are language-independent. The developed applications can be deployed and run on any system without considering the Blockchain core system.

Sawtooth has a built-in parallel scheduler that divides the transaction into smaller ones. These sub-tasks are executed parallelly by different nodes while maintaining transaction integrity. In addition, the Sawtooth also prevents the double spending attack with the presence of simultaneous modification to the transaction. practical Byzantine Fault Tolerance (pBFT) and Proof of Elapsed Time (PoET) are the two mostly used consensus mechanisms in this platform.

Table 1.1: Summary of Hyperledger framework

Hyperledger Framework	Consensus Mechanism	Key Feature	Supported Language for Smart Contract
Burrow	PoET, BFT	To execute the Ethereum Virtual Machine (EVM) based smart contract in a permissioned environment	Solidity
Fabric	PoS, DPoS, pBFT	Private Data Collection	Javascript
Indy	BFT	Decentralized Identifiers or DID	Python, Node.js
Iroha	Crash Fault, YAC	Requirement of multiple signatures from different nodes to validate the transaction	C++, Java
Sawtooth	pBFT, PoET	Development of platform-independent application	Language Independent

(iv) **Corda:** Corda is a permissioned Blockchain platform. Unlike other platforms, this system only shares data with users associated with a particular transaction, rather than sharing it with the entire network. The emerging functionality such as "*No Block*, *But Chain*"makes it different as compared to the other platforms. In "*No Block*, *But Chain* the platform creates a chain of transactions which are dependent on each other, instead of keeping the hash of the previous block. This functionality enables the platform to optimize the storage requirement in the network. Corda does not batch up the transactions and confirm them all at once like the other Blockchain platforms. Corda verifies each transaction as it occurs. Byzantine Fault Tolerance (BFT), and pBFT are two mostly used consensus mechanisms

used in Corda.

Table 1.2 shows an overview of the different Blockchain platforms discussed above. Different parameters including consensus mechanism, transaction scalability, currency, Blockchain types, etc are considered for comparing the discussed Blockchain platforms.

Table 1.2: Summary of Blockchain frameworks

	Bitcoin	**Ethereum**	**Hyperledger**	**Corda**
Consensus Mechanism	PoW	PoS, PoW, DPoS, PoA	PoS, DPoS, BFT, pBFT, YAC	BFT, pBFT
Currency	Bitcoin(BTC)	Ether (ETH)	NA	NA
Smart Contract	Yes	Yes	Yes	Yes
Language for Smart contract	Clarity	Solidity	Node.js	Java
Transactions per Second (TPS)	7 TPS	30 TPS	3000-20000 TPS	15-1678 TPS
Type	Public	Public	Private	Private
Turing Completeness of Smart Contract	Incomplete	Complete	Complete	Complete
Governance	NA	Decentralized Autonomous Organization (DAO)	Linux	R3
Mining Process	Allowed	Allowed	NA	NA

1.3.4 Smart Contract

It is a collection of code and data which is executed in a network during a transaction. Although different nodes can execute the smart contract, the result of the execution should be consistent and stored in the distributed ledger technology (DLT). Smart contracts can perform computations and store information. They must be deterministic, meaning they produce consistent output when given the same input conditions. Smart contract implementation in a Blockchain makes the network autonomous, allowing it to trigger automatically upon reaching predefined conditions.

A smart contract is consist of "if" and "$then$" statements. As soon as certain circumstances are satisfied and confirmed, the activities are carried out by a network of computers or nodes. These measures may include distributing payments to the correct individuals, registering a vehicle, sending out alerts, or issuing a citation. When the transaction is finalized, the Blockchain is updated. It indicates the transaction is final and only those with access may read the details.

A smart contract includes multiple conditions to ensure all active nodes of network to be satisfied with the result of the transaction. Based on the "if and then" rules in the smart contract, all nodes must decide how the transaction and entire blocks will be recorded in the DLT. During the selection of CA for a transaction, the smart contract leads to a common decision. The main advantages of using the smart contract in a transaction are as follows:

- Efficient and accurate.
- Transparent
- Secure

1.3.5 Consensus Mechanism

The main building block of the Blockchain is the consensus model. It is the fault tolerance mechanism adopted by the Blockchain for obtaining an agreement on a single value among all participating nodes. It has two basic features such as security, and fault tolerance. The security part ensures that all transactions must be recorded in the Blockchain. Fault tolerance allows the network to continue processing transactions by disregarding the faulty participating nodes. The main objective of this principle is to select a validator or certificate authority (CA) for validating the individual transaction. Various popular consensus algorithms are elaborated below.

- **Proof of Work (PoW):** Nakamoto's Proof of Work (PoW) is the most widely recognized consensus mechanism, and it is utilized by Bitcoin. For a long time, the Proof of Work has served as a reliable technique for monetary cryptography. The computer or node executes a large amount of computations to conclude an answer of a mathematical problem. In order to solve these enigmas, the hash function is used. Hash is a complex mathematical method which verifies the authenticity of blocks containing transactions. In a nutshell, the data in a block consists of the hash of the prior block, the timestamps of all the transactions inside a block, a nonce, and the hash of the current block. A node called *miner* attempts to solve the puzzle by searching a specific nonce value. This nonce must satisfy a predefined requirement. For example in bit such as turning the first 30 bits of the hash value into zero. The network's scalability and adaptability to new situations are greatly enhanced by the ability to modify these parameters. In Proof-of-Work, a hash value of the block header is computed by each node in the network. To put it another way, miners strive to discover hash values that are equal to or less than a predetermined number in order to achieve network consensus. When a node in the network reaches the desired value, it will send a broadcast to the rest of the network, and the other

nodes will need to verify the validity of the hash value. For this reason, if the block is genuine, all nodes would add it to their respective chains.

- **Proof of Stake (PoS):** It is just the advanced version of the PoW which removes computational limitation of nodes. PoS allocates a identical computational capcity to all miner nodes for solving the puzzle. Therefore, this model creates an opportunity to thin client nodes for becoming validator of key pairs and the digital certificate of transaction.

- **Deligate Proof of Stake (DPoS):** DPoS implements a delegate election mechanism for key pair and certificate verification. This consensus model, select a group of participating nodes known as delegates before initiating any transaction. The node present exclusively in the delegate group can validate the transaction. The nodes from this group are chosen as a CA in such a way that every node has an equal chance to validate the transaction. In addition, the delegates or nodes are changed after a defined time slot. The main limitation of this algorithm is the semi-decentralization concept which is very hard to implement in a Blockchain network.

- **Byzantine Fault Tolerance (BFT):** It allows to reach a common value considering certain number of faulty participating nodes in the network. The primary objective of this consensus model is to provide robustness to the Blockchain network. BFT can tackle miner nodes which generate malicious information during the transaction.

- **Practical BFT:** In PBFT when a CA receive a transaction for verification, it immediately sends a query to all available miners in the network about the correctness of the received block. Thereafter, it waits for the reply from other nodes. If majority of participating nodes verify the transaction, then it will accepted in DLT. If the maximum of miner nodes gives a positive result regarding the correctness, then the transaction block is considered and verified otherwise the initiator node is declared as the malicious one. This information is forwarded to all participant nodes. In future, if any transaction is issued by that malicious node, it will be rejected. Generally, Hyperledger technology adopts this kind of consensus model.

- **Proof of Authority (PoA):** This algorithm depends on the reputation of a participating node. This model first verifies the identity and the behavior of the participating node before making it a validator or CA. Once the transaction is completed successfully, the reputation of the node is increased by one. The higher the reputation of a node, the greater the probability of becoming a validator or a CA for upcoming transactions. If the transacton is unsuccessful, the reputation will

be decreased. The node which has the highest degree of reputation is chosen as a validator of key pair and transaction.

- **Proof of Activity (PoAc):** It is combination of both PoW and PoS. A group of nodes is elected as the validators. For each transaction signatures from all elected nodes is required. Once all the nodes of the group sign, the transaction is considered as successful otherwise it declared as an unsuccessful. The main limitation of the PoAc is computational overhead. Decred : citation is the only cryptocurrency using the PoAc as the consensus model.

- **Proof of Capacity (PoC):** PoC depends on the storage space of nodes. Before starting the mining process, every node need to store a data known as plot in their corresponding storage. The node with more plots in its storage space have more probabilty of becoming a validator for a transaction. However, the main limitation of PoC is storage capacity. Currently, Burstcoin is the only cryptocurrency that is using PoC as the consensus model.

1.3.6 Public Key Infrastructure (PKI)

Public Key Infrastructure (PKI) is a technology used to authenticate devices or nodes or users in internet-based communication. It employs asymmetric key for encryption, digital signature for authentication and hashing to preserve data integrity. PKI encompasses hardware, software, rules, and methodologies that facilitate the creation, management, and storage of digital certificates to enhance transaction security. The primary concept involves one or multiple nodes which are responsible for issuing digital certificates, along with the corresponding public and private key pairs. PKIs are valuable because they can verify the authenticity of users or nodes and services, allowing for granular access to data. As reliance on internet-based communication continues to grow, so does the demand for authentication and compliance with robust data security regulations. PKI is rapidly gaining prominence as a preferred solution for next-generation applications that demand robust authentication and advanced encryption for heightened security.

PKI provides an environment where cryptographic security mechanisms such as digital certificates and digital signatures are widely used to secure data. PKIs ensure the confidentiality, integrity, and availability (CIA) properties of transactions between different nodes using asymmetric cryptography. In PKI context, nodes refer to various entities involved in internet-based communication, including individual end-users, web servers, embedded systems, linked devices, and programs/applications. Each node within the network must possess a private and public key pair. The private key is exclusively used by its owner for revoking digital signatures, while the public key is utilized by other network nodes to verify transactions.

PKI is used in secure socket layer (SSL) / transport layer security (TLS), Internet Protocol Security (IPSec) and many other applications. As electronic transactions, digital documents, and the number of connected devices on the internet increase, the significance of PKIs extends beyond standalone solutions such as secure email or encrypted online traffic. Moreover, PKIs are responsible for preserving trust in communication. The primary use case of PKI can be stated as follows:

- Email security.
- Web browser security.
- Encryption and decryption of the data.
- Digital signing of the software and applications.
- User authentication.

1.3.6.1 Building blocks of PKI

The main building blocks of PKI are *digital certificate*, *Certficate Authority* (CA), *Registration Authority* (RA), and *Validation Authority* (VA).

- **Digital Certificate:** Digital certificates are required to verify the identity of nodes participating in a transaction. The Certificate Authority (CA) is responsible for issuing digital certificates. These certificates have two primary functions: one is to identify the users associated with the transaction, and the other is to encrypt and decrypt messages between sender and receiver. To ensure the credibility of a node in a specific platform, digital certificate must be reliable.
- **Certficate Authority:** A Certificate Authority or CA, plays a crucial role in a PKI which establishes a trust hierarchy. CAs are responsible for distributing digital certificates, which are used to authenticate the identity of users. In one word, CA is the foundation of a PKI security. The growing use of PKI in various applications also raises the likelihood of targeted attacks on the Certificate Authority (CA). Therefore, physical and logical restrictions, along with hardware security modules (HSMs), are necessary to maintain the authenticity of PKI.
- **Registration Authority:** The role of the Registration Authority or RA in PKI is to accept the certificate request of different nodes. After receiving the request it verifies the requester's credentials . The features of a RA include authenticating the node's identity, determining th status of certificate issuance process and executing the node's request to revoke the certificate. Certificate assesment and distribution are the exclusive responsibilities of RAs in PKI.

- **Validation Authority:** The Validation Authority or VA in PKI validates the certificate issued by the CA. Since certificates can be both granted and revoked, it is essential to verify their authenticity before placing any reliance on them. The VA plays a key role in this process. The public key of the newly created certificate is supplied to the VA by the issuing CA. After receiving the certificate verification request from the receiver, the VA uses the stored public key to authenticate the certificate.

- **Certificate Revocation List (CRL):** A Certificate Revocation List (CRL) is a collection of certificates that have been issued by a Certificate Authority (CA) but have been subsequently revoked by the same CA before the certificate's expiration. *Delta* and *Base* CRL are the two different variants of CRL. *Base* CRL is a large list that contains entire revoked certificate details and *Delta* CRL contains the most recent list of revoked certificates. In each short time interval, the $DeltaCRL$ is updated to remove the older revoked certificates.

- **Hardware Security Model (HSM):** It is the optional element for the PKI that helps in safeguarding the key pairs.

1.3.6.2 Working principle of PKI

The Public Key Infrastructure (PKI) comprises hardware, software, and cryptographic rules to create, store and manage digital certificates for secure internet-based communication. When a sender wants to initiate a transaction, it generates a digital certificate generation request and sends it to the VA present in the system. Upon getting the certificate request from the user the VA verifies the identity of the sender. Once this verification process is complete, it forwards the request to the CA. After getting the request from the VA, CA generates the certificate along the key pair (public and private key). The generated private key is shared with the certificate-requesting user, and the public key is shared with the VA for the certificate verification in the future. The VA stores the received certificate public key in the CRL.

The sender encrypts the data along with the certificate by using the private key shared by the CA. The receiver after receiving the data forwards it to the VA for certificate verification. VA invokes the public key from the CRL to verify the certificate. Upon successful certificate verification, the VA informs the receiver regarding the authenticity of the certificate and then only the receiver decrypts the data otherwise the transaction is discarded.

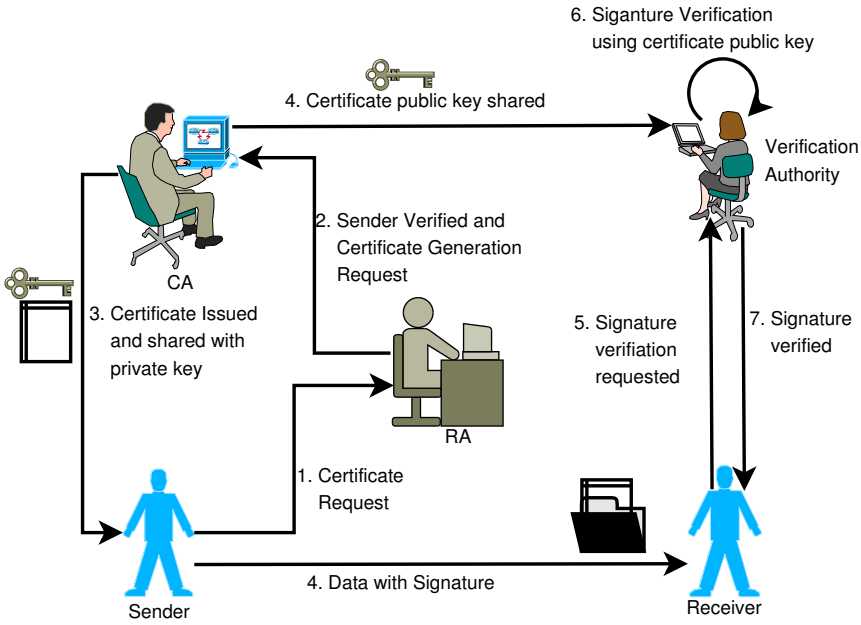

Figure 1.4: Working of conventional PKI system

1.3.6.3 Limitation of PKI

Undoubtedly, the traditional PKI system is one of the most popular security solutions for many applications. However, PKI suffers from several limitations, which make it inefficient. Three major limitations are as follows:

- Entire application system depends upon one or more CA(s) for any kind of transaction. The number of CA is very small compared to the number of the users or nodes. As the number of users or nodes increases, the load on the CA to issue certificates also increases. As a result, the performance of the CA can start to degrade, and the entire infrastructure can suffer from low performance.

- There is no certain way to identify the validity of the CA. If the CA starts providing a malicious certificate then the application has to rely on that which makes the entire network malicious.

- The degree of the successful transaction between the client and server depends on the correctness of the certificate issued by CA. The application system based on the conventional PKI system relies on the third-party centralized CAs. If it becomes malicious, (citation required for malicious CA) then the entire communication will be compromised, and it leads to a single-point failure.

- The conventional CA is time-consuming because numerous amount of users or nodes can choose a single CA.

1.3.7 Blockchain-based Public Key Infrastructure

The limitations discussed in section 1.3.6.3 make it difficult to adopt the conventional PKI system as the primary security solution of different applications. To address the aforementioned limitations, there are initially two possible solutions:

1.3.7.1 Log-Based PKI (LB-PKI)

Log-Based PKI (LB-PKI) uses publicly available log servers for monitoring and publishing the digital certificates issued by the dedicated CA. These public logs provides transparency to the end user so that the misconfiguration in the certificate can be noted down by the client as well as the server. The only constraint present in the model that makes it more applicable is that the end user only accepts and trusts certificates that have been publicly logged. Google Certificate Transparency is a well-known, currently deployed LB-PKI that is in use in both Google Chrome and Mozilla Firefox

1.3.7.2 WoT or Web of Trust

WoT or Web of Trust is a fully decentralized PKI framework where the end users or nodes can select a trustworthy party who will sign the public key certificates. Pretty Good Privacy (PGP) encryption model uses WoT where one known user or node signs the certificate of another user or node. While a new node joins the network, the signing or validator node of this transaction is always unknown to the newly joinig node. To create mutual trust between the signing node and newly joining node may require significant amount of delay. Unlike LB-PKI, certificate revocation is not possible if a node in WoT loses its private key. In LB-PKI, the node who lost its private key appoint another node as a certificate revoker.

Blockchain has solved the major issues reported $LB-PKI$ and WoT. Blockchain is decentralized, immutable, secure, and rulled by smart contract which make it suitable for various applications. The use of blockchain in PKI can handle issuance and revocation of certificate effciently and it eliminates the single-point failure problem. It has an advantage over the WoT model because it eliminates the requirement for trustworthiness participant nodes in a large network. The blockchain-based PKI does not require trustworthy members for signing the public key certificate. It has its own way to select a CA for every transaction which solves the issues of the WoT approach reported earlier.

The blockchain-based PKI achieves more attention for its distributed trust and distributed log of transactions to verify the activities of the CA. It solves the issue of centralized public log used in LB-PKI. For every transaction the blockchain-based PKI needs to select a different CA which eliminates the limitation of single-point failure reported in conventional PKI. Considering the above-mentioned reasons, Blockchain-based PKI becomes an emerging alternative of conventional PKI.

1.3.8 Blockchain-based Applications

Recently, blockchain technology has garnered significant attention from both academia and industry. The features of blockchain, including decentralization, immutability, and transparency, enable various application domains to use blockchain as an emerging technolog for developing the decentralized applications. Blockchain is implemented in a peer-to-peer network to develop decentralized applications, and distributed ledger technology (DLT) is used to store all possible transactions of a blockchain-based application. This DLT is accessible to all active participants in the network. The applications of blockchain technology can be broadly categorized into two main categories: those in the financial industry and those in the non-financial sector [18]. Different application domains of blockchain technology include supply chain management (SCM), the Internet of Things (IoT), healthcare, governance, finance, education, etc. Figure 1.5 depicts a variety of Blockchain-based applications that are diverse in nature.

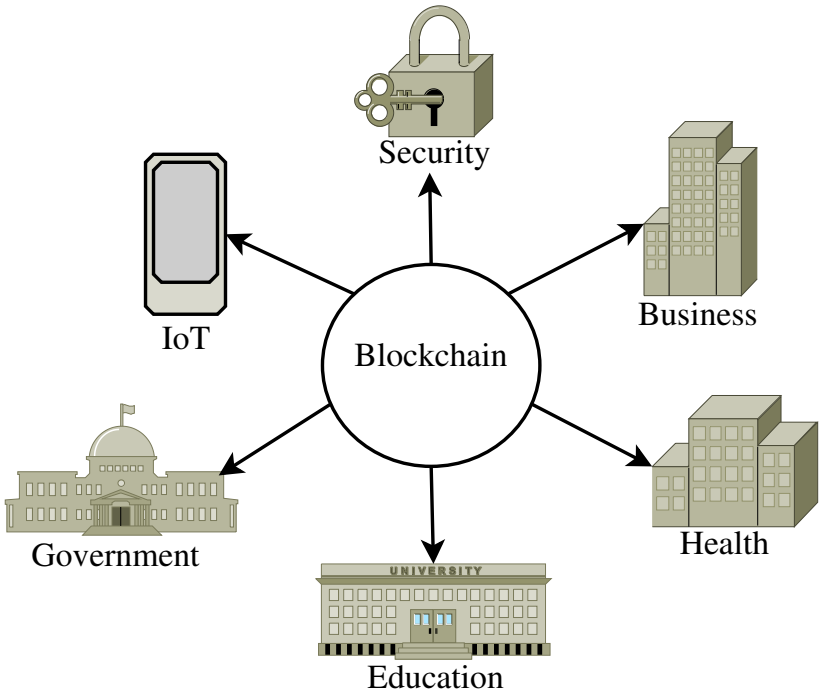

Figure 1.5: Usecases of Blockchain technology

- **Internet of Things (IoT):** In recent years, IoT devices have generated around 95 percent of all global information. The emphasis on decentralization has presented an opportunity for the use of Blockchain technology to enhance the security of IoT devices [13].

- **Business:** Currently, Blockchain technology is used in a variety of business areas, including the financial and non-financial sectors, to improve the efficiency of the platform. It is widely believed that Blockchain technology will play a significant

role in driving the growth of the global economy in the coming years. Blockchain 1.0 refers to the initial stage of Blockchain technology, primarily represented by the cryptocurrency Bitcoin, which has gained significant attraction in the financial industry. Blockchain technology is currently becoming more significant in various business sectors, such as sales, the claims process, and the payment process, among others.

- **Security:** The primary concern with centralized applications is the risk of a single-point failure. An attack on the central authority responsible for data storage may severely impact the application system's performance. However, blockchain technology provides both decentralization and immutability, which eliminates the shortcomings of conventional systems. [19].

- **Healthcare:** The EMR or electronic medical record is the most significant use of information technology in the healthcare system. Because of the DLT, it is possible to retain a unique and secure record of a patient. It will include all forms of data for an individual patient including various test reports, a list of pharmaceuticals during the treatment, prescription information, and so on. This information is stored in DLT for better accessibility without considering the region boundary.

- **Education:** Blockchain technology offers a secure and private platform for storage and management of data in a ubiquitous learning environment, addressing concerns related to security, privacy, and storage. In addition, blockchain serves as a safety mechanism for the collection, storage, and analysis of academic data.

- **Governance:** Blockchain technology can be applied to various systems, such as marriage registration, patent management, and tax handling, that involve public records and data.

1.4 Motivation

Blockchain is usually famous for its critical role in cryptocurrency. However, it is also popular for other applications such as : e-governance, supply chain management (SCM), healthcare, finance, etc. Blockchain can guarantee the fidelity and security of data records and avoids the need for a third party due to its pervasive features such as immutability, decentralization, and distributed ledger technology (DLT).

However, there are some issues present in this technology. Two of the most important problems are security and computational overhead. Blockchain network simply avoids the inclusion of third party or middle man for performing any kind of transaction. This feature resists man-in-the-middle (MITM) attack on Blockchain network. Other most common

type of attacks on the Blockchain network are denial-of-service (DoS) and distributed DoS (DDoS) attacks. However, there may still be other forms of attack resistance that have yet to be discovered within the realm of Blockchain technology. DoS and DDoS attacks aim to overload a particular peer or multiple peers within the network with spam transactions, potentially causing a specific number of nodes to go down.

However, the Blockchain network does not rely on specific nodes to function. Instead, all peers have copies of the distributed ledger, which makes the Blockchain network resistant against single point failure issues. Most of the present Blockchain-based PKI solutions such as $Block-CAM$, $BC-Trust$, $PB-PKI$ citatios??????, etc, provide a way to deal with the above mentioned attacks. Apart from DoS and DDoS attacks, there are numerous other types of attacks that are relevant to Blockchain networks and need to be thoroughly explored.

On the other hand, each blockchain transaction needs a CA selection process. The blockchain network adopts consensus algorithms to select the CA for each transaction. This selection process requires huge computational overheads. This CA selection process can cause a long dealy and a heavy computational overhead in a scaled Blockchain network,.

1.5 Objective

The primary objective of this thesis is to explore different blockchain-based application which are divided into two primary parts, as depicted in Figure 1.6. The first part is to design and develop a blockchain-based application to store and share different types of data. As a proof of concept, the author used electronic health records as data in a blockchain-based platform for communication between doctors and patients. The second part is to design and develop two blockchain-based PKI solutions which aims concerns : smart contract, consensus algorithm, network computation overhead, cyber attacks, validator selection process, trust of nodes etc. The contribution of this work can be summarized as follows:

- Design and development of a basic blockchain-based decentralized application for storing and exchanging data like EHRs between doctors and patients. This is a basic implementation to understand working principle of Blockchain-based applications.
- Design and development of blockchain-based PKI system which uses secure peer-to-peer (P2P) communication. The proposed blockchain-based PKI utilizes a smart contract to preventmany common threats : DoS, DDoS, MITM, 51% attacks, injection attacks, routing attacks, and Eclipse attacks. This blockchain-based PKI provides equal opportunity for all available nodes to become Certificate Authority.

- Design and development of a PKI which uses clustering approaches based on validation time, response time, and trust. The consensus algorithm utilized in this work searches for nodes within the chosen cluster rather than searching through all participant nodes. This helps to reduce the search space of the Certificate Authority selection process.

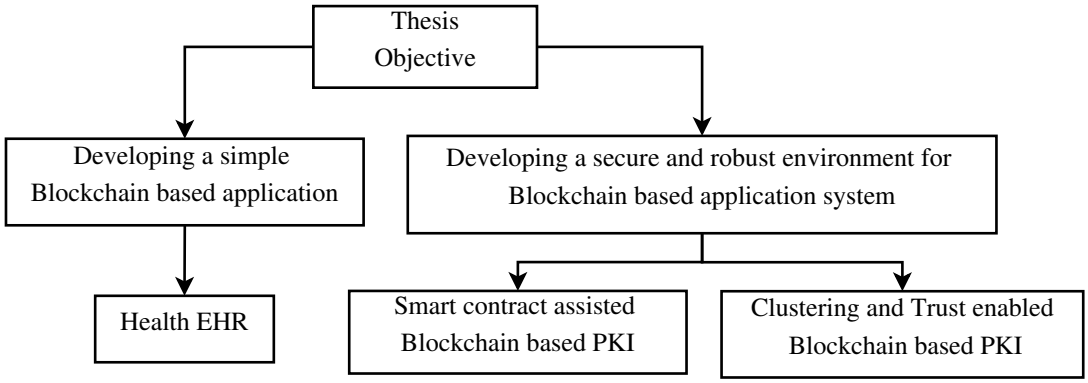

Figure 1.6: Objective of the Thesis

1.6 Thesis Ogranisation

As illustrated in Figure 1.7, this dissertation is organized as follows.

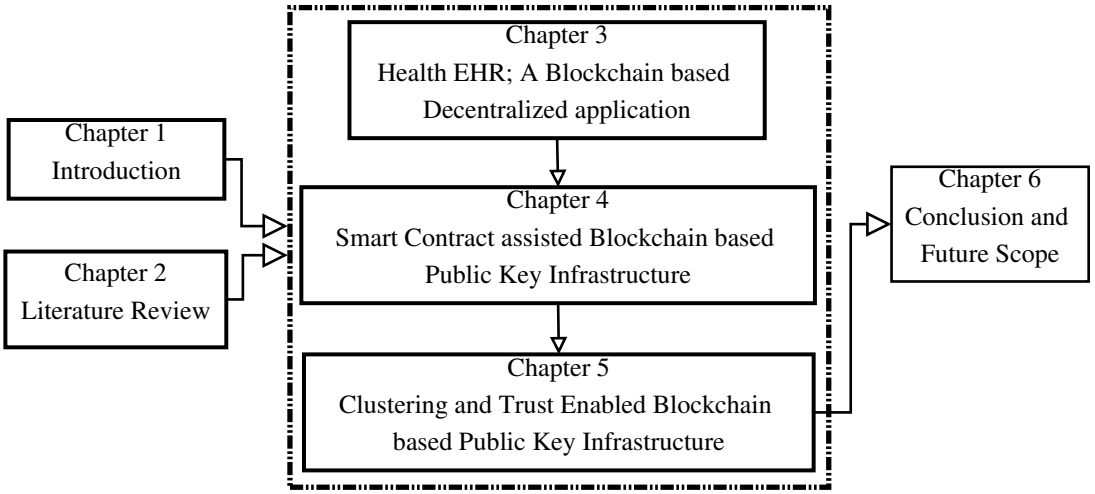

Figure 1.7: Stucture of the Thesis

Chapter 1 provides the context, motivation, and objective of the thesis. Additionally, it presents a preliminary study of Blockchain technology, including the working principle of Blockchain, different Blockchain platforms, and various consensus mechanisms available in this technology.

Chapter 2 provides a brief literature review of the dissertation, which is divided into two parts. The first part reports on the application of Blockchain in various domains,

including the Internet of Things (IoT), governance, healthcare, supply chain management (SCM), and finance. The second part of this section provides a brief review of Blockchain security, including the Blockchain-based PKIs.

Chapter 3 presents the first contribution of this thesis, in which a basic Blockchain-based decentralized application is developed. The primary objective of this application is to provide a platform for storing and sharing data generated from doctors and patients in EHR.

Chapter 4 presents the second contribution of this dissertation, in which a Blockchain-based PKI is developed. The main focus of the developed Blockchain based PKI is to prevent various attacks, including DoS, DDoS, MITM, 51%, Injection, Routing, and Eclipse attacks. Additionally, the developed Blockchain-based PKI provides an equal opportunity for all available nodes to become Certificate Authority (CA). The developed Blockchain based PKI is evaluated based on gas utilization, time-lapse for key generation, and validation process.

Chapter 5 is the final contribution of this work where another Blockchain-based PKI is developed to reduce the searching space CA selection using K-Means and DBSCAN clustering algorithms. This PKI is evaluated over Response Time and Validation Time in contrast to PKIs without clustering. Additionally, gas utilization parameter is used to evaluate the performance of the proposed PKI.

Finally, Chapter 6 holds the overall conclusion and future scope of this thesis.

Chapter 2

Related Work and Research Gap

This chapter shows the detail study of various blockchain-based applications. This article explores various attacks on blockchain networks and how PKI-based security solutions can mitigate them. Figure 2.1 represents the detailed structure of this literature survey. Section 2.1 shows the applications of blockchain in various usecases, section 2.2 represents the application of blockchain in security.

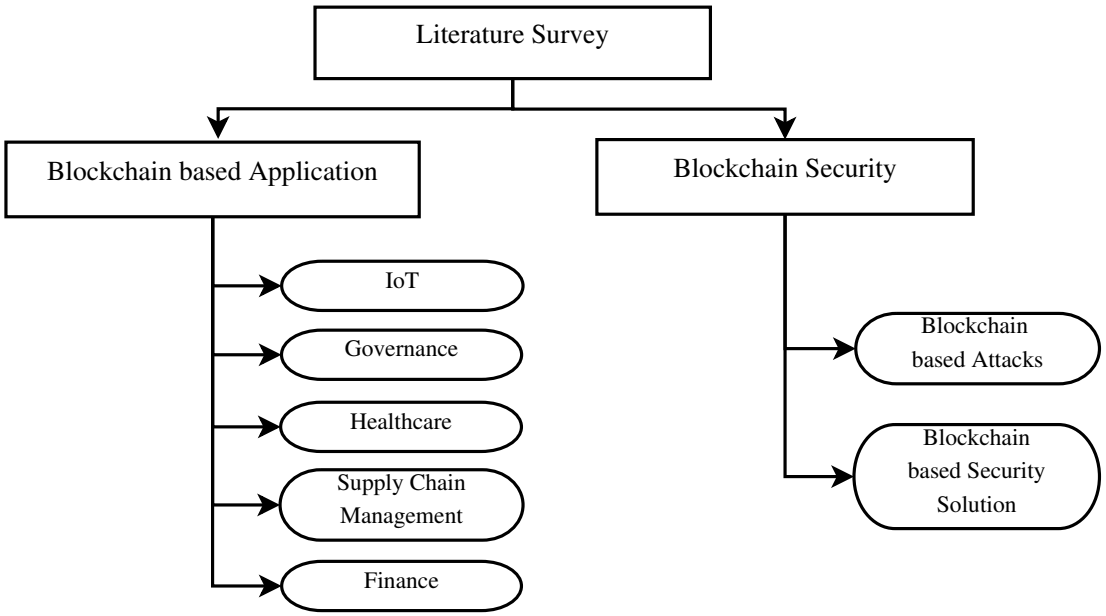

Figure 2.1: Literature Survey detail layout

2.1 Applications of blockchain in various usecases

This section shows a extensive study of blockchain applications in various domains : IoT, Governance, Healthcare, Supply chain Management (SCM), and Finance etc.

2.1.1 IoT

In order to solve the issues with data centre-based storage, simple data Lu et al. [17] proposed a blockchain and IoT based solution to deal with the issues of conventional food anti-counterfeiting prevention methodologies. The issues of the conventional system are limited strorage capacity, data silos. Traceability data of food is stored throughout the food manufacturing, sale, and transportation processes using blockchain technology. The decentralised and immutability, ensuree the food's authenticity.

To detect the distributed denial of service attack in IoT enabled network Kumar et al. [18] proposed a novel approach using blockchain and fog computing. The performance of the proposed model is evaluated by implementing the Random Forest (RF) and Xtreme Gradient Boosting (XGBoost) machine learning technique. The main objective to provide a prediction of DDoS attack to the IoT network.

In [19], Yin et al. presented the SMARTDID system, a blockchain-based innovation that would assign a unique identifier to each device in an Internet of Things network. The primary goal of this approach is to improve the security of the linked devices in terms of their private data. Since logistics data may be acquired through IoT sensors, Ugochukwu et al. [20] presented a Blockchain-based IoT-enabled system architecture for safe and effective logistics management. The author has created and detailed the sequence diagram for a smart contract that encrypts all communications between stakeholders involved in logistics.

Powel et al. [21] presented a method to restrict the potential of arbitrary claims on the performance of IoT data, and; to leverage mechanism design as a technique that may be used to incentivize supply chain behaviour that enhance the likelihood of desirable eventualities being achieved.

Bandara et al. [22] propose a minimal blockchain platform called "Tikiri" for low-power Internet of Things gadgets. It employs Apache Kafka as its consensus mechanism and suggests a novel blockchain architecture to deal with the execution of transactions in real time. It has a smart contract based system for allowing for the parallel execution of blockchain transactions.

Raghav et al [23] proposes a novel consensus mechanism PoEWAL for blockchain-based IoT (BIoT) network. The proposed consensus mechanism aims to reduce the computational overhead caused due to the PoW consensus mechanism. The main advantages of using this consensus mechanism is the requirement of low mining time.

Identity management is the key feature of IoT based networks for controlling access to the IoT device generated data. Conventional mechanism used to control the access employs a centralized controller which is pruned to single point failure. Bouras et al. [24] proposed a blockchain based identity management system for managing the distributed access to the IoT data. The proposed system ensures the confidentiality, integrity, and

availability (CIA) characteristics of the IoT device generated data.

Table 2.1: Summary of applications of blockchain in IoT

Reference	Application System	Key Functionality
Lu et al. [17]	Food Uniqueness Management System	The system uses blockchain technology's decentralized storage and interoperability to retain food traceability data throughout manufacturing, sale, and transit to assure food uniqueness.
Kumar et al. [18]	Security	A platform to detect the DDoS attack in a IoT based network.
Yin et al. [19]	Privacy Preserving System	A Blockchain-based system has been developed for identity anonymity of all connected IoT devices. In addition the proposed system helps in maintaining the privacy of the on-chain data.
Ugochukwu et al. [20]	Logistic Management in Industry 4.0	Blockchain-based IoT-enabled system framework for secure and efficient logistics management.
Ahmed et al. [22]	Smart City	A smart and sustainable conceptual framework that leverages cloud computing, IoT devices, and artificial intelligence to process and obtain necessary information.
Bandara et al [23]	Light weight IoT Device	A lightweight blockchain platform has been developed to support the parallel transaction between the IoT connected lightweight devices.
Bouras et al. [24]	Identity Management	Provides a novel blockchain based approach to control the access to the IoT device generated data.

2.1.2 Governance

Lee et al [25] developed a blockchain based identity authentication mechanism (BIDaaS). The proposed system is developed for authenticating the mobile telecommunication subscriber identity management for better accessibility towards the stored data. The primary concept in article [26] is to realise the decentralised e-voting application without an impartial third party using blockchain technology homomorphic encryption. This is a verifiable voting procedure that ensures voters' anonymity, privacy of data transfer, and verification of votes before they are counted.

Sullivan et al. [27] proposed a blockchain based identity management service for Estonian government. E-Residency is a revolutionary concept where Estonian government has formed a partnership with Bitnation to provide a public notary service to Estonian

e-Residents using blockchain technology. The use of blockchain technology in electronic residency programmes has the potential to bring about significant changes in how identification information is monitored and verified.

Maura et al. [28] proposed a blockchain based voting system to avoid the transparency issue present in the traditional voting system. The primary objective of this work is to provide a decentralized infrastructure along with the public key cryptography for sharing the voting related data. In addition the voter's identity anonymity is another feature of this voting system.

Electronic voting using a permissioned blockchain has been developed by Hjalmarsson et al. [29]. They have deployed a private Proof-of-Authority (PoA) blockchain built on top of Go-Ethereum to achieve privacy and security. The identity as a stake consensus approach allows for faster transaction delivery. District nodes and Bootnote nodes are the two types of nodes used in their implementation. District nodes represent voting districts and manage smart contracts, whereas bootnotes represent institutions with private network access and act as intermediaries between district nodes.

Hau et al. [30] proposed an application system for e-government services in China. The main objective of this application is to improve the quality of the government services. In addition, it also focuses on efficient sharing mechanism of government data with high transparency. However, the security, and reliability are two major drawbacks of the proposed system.

According to Khan et al. [31], blockchain technology could enable the full integration of e-business and e-government services, which would streamline and strengthen government operations. The authors of this research investigate the development of an e-government service in Dubai, United Arab Emirates. To facilitate the creation and updating of vehicle licence information through the blockchain, the developed sultion implements Unified Corporate Registry (URC). The register is connected to public databases and private companies. Whenever a license is issued, renewed, changed, or cancelled, the license information is automatically inserted from each node to the associated business activity. Members of the unified registry fall into one of three categories: a) the users uploading the license to the registry, b) users subscribing the license, and c) the users controlling and securing the access to the stored licenses.

Paez et al. [32] proposed a blockchain-based framework for providing a digital identity to the citizen of a country. The main objective of this framework is to manage the transactions between different users or nodes. Every user is authenticated by the unique id provided by the framework and their transactions are validated by using the biometric and iris recognition. There are two types of users present in this framework: a) user who generates and validates digital certificate along with the key pair: b) the users who are doing some kind of transaction.

Liu et al. [33] suggested a framework for exchanging data across government agencies

without breach of data. Private blockchains provide information exchange between nodes, authenticating the nodes on the network so that they may trust one other. Simultaneously, it reduces data framework instability and sets up the data's core properties. The system not only collects the names of the departments that possess the data but also enables them to exchange requests based on specific criteria. Furthermore, the blockchain is capable of sorting aggregated user messages anonymously and processing user data in a confidential manner.

Ghanem et al. [34], proposed a blockchain based framework to form an interoperable network between participating entities to enable e-services offered via the e-government portal. This paradigm improves the flow of data between government agencies, businesses, and citizens. Citizens may safely and securely make service requests through the e-government site using their issued e-Key. The upkeep of e-Keys is the responsibility of the people.

Zhang et al [35] proposed a framework using blockchain for sharing the data between the various government sectors such as education, health, tax, and legal departments. PoW consensus mechanism is used to sync the data in DLT. The data needs to be hashed by using SHA-256 hashing function. The developed platform is responsible for converting ciphertext back to plaintext and includes a proper mechanism to validate the end users of a transaction.

Nguyen et al. [36] has proposed a blockchain application for issuing land value certificates. Since all data centres must be located inside Vietnam under the Vietnamese network security regulation, a permissioned blockchain is appropriate for this developed application. All of the natural resources are identified by the developed framework through the e-governance application. The proposed work has three different layers. The first layer of the framework is concerned with a dApp for managing the valuation certificates. The second layer of the framework is used to store the valuation certificate in the blockchain network. The final layer is used for tracing the status of the stored certificates. However, complex structure of this application is expensive to adopt.

2.1.3 Healthcare

Huang et al. [37] proposed a blockchain based healthcare application to detect the data modification in the stored data. All use cases of the healthcare domain including the hospitals, patients, and doctors uses the blockchain platform to store their data. Additionally, the attribute-based re-encryption mechanism has been used to ensure granular access to the stored data.

The conventional healthcare system faces isssues in security, privacy, transparency, and authorized access. Zhuang et al. [38] proposed a blockchain-based healthcare application in British Columbia that relies on smart contracts to govern data storage and access.

Table 2.2: Summary of applications of blockchain in IoT

Reference	Application System	Key Functionality
Lee et al [25]	Identity Management System	A blockchain based system for identity management in a mobile subsriber organization.
[26]	E-Voting	A platform for E-Voting System while keepig voters' identities secret, keeping data transfer private.
Sullivan et al. [27]	E-Residency	A Blockchain-basedid management service for Estonian government.
Maura et al. [28]	E-Voting	The primary objective of this work is to provide a decentralized along with the public key cryptography for sharing the voting related data.
Hjalmarsson et al. [29]	E-Voting	A blockchain based framework for developing the e-voting system while keeping the voter's identity anonymous.
Hau et al. [30]	E-Government service	It focuses on efficient sharing mechanism of government data with high transparency.
Khan et al. [31]	E-Business and E-Government service	Blockchain is used to accomplish complete integration of e-business services and e-government services.
Paez et al. [32]	Identity Management	A blockchain-based framework for providing the digital identity to the citizen of a country.
Liu et al. [33]	Information Sharing	a framework for exchanging data across government agencies with a particular emphasis on preventing data breaches.
Ghanem et al. [34]	E-Goverment Service	A blockchain based framework to enable e-services offered via the e-government portal.
Zhang et al [35]	Information Sharing	A framework using blockchain for sharing the data between the various government sectors such as education, health, tax, and legal departments.
Nguyen et al. [36]	Land valuation certificate management	A blockchain-based, e-government services-integrated, generic model for storing the land value certificates.

Patient oriented data exchange is achieved by customising data segmentation and creating a "endorsed list" of doctors with access to the data.

Guo et al. [39] proposed a blockchain based model to control the access of health-related data. Off-chain storage is a preferred solution for the healthcare industry due to

the large size of data. This literature adopted a blockchain-based controller which can control all the access of corresponding data. In addition, this model maintained a count of every access of the storage. The smart contract has a threshold value for accessing the information, and users who exceed the limit are strictly prohibited from accessing it in the future.

To ensure the privacy of the health information of patients, Kim et al. [40] developed a blockchain-based, trustworthy solution. Electronic health records for patients are maintained on cloud, where the data is protected by transaction tracking and access control. The health records stored in the cloud were encrypted using an elliptic curve cryptosystem.

Chenthara et al. [41] presents a dApp for British Columbia with an abjective to main the EHR privacy. This framework provides an effective and scalable platform to store the health data which maintains privacy and security. Interplanetary File System (IPFS) is the foundation of this proposed health-chain. The public key encryption algorithm is used to encrypt the data before storing in IPFS to ensure the CIA of stored data.

When electronic health record data is compromised, patient confidentiality is also compromised. Health data can be protected against manipulation, and data sharing can be simplified with the use of blockchain technology. Chen et al. [42] introduced a BC-based EHR encryption which enables users to search the information through a specific set of indexes stored in the blockchain. This approach safeguards the authenticity, and trackability of the index.

Conventional health insurance systems rely on a centralized design. The centralized system requires regular human involvement to verify and process insurance claims. The centralized system is prone to single-point failure. To avoid single point failure issue, Karmakar et al. [43] proposed a blockchain based dApp for managing insurance claims. A smart contract is deployed with three different functionalities including verification, insurance processing, and claiming status recording. When the application receives an insurance claim request then the smart contract is used to verify the identity of claimers. After successful verification, the smart contract is invoked to process the insurance claim. Finally, the status of the claiming process is recorded in the blockchain network using the smart contract. However, the dependecne on crypto wallet used in this work increases computational overhead.

Saldamli et al. [44] proposed a blockchain based for securing the healthcare insurance system. The proposed system employs a smart contract to detect the frauds occurring to the present insurance system. It is achieved by imposing some accessibility mechanism to limit the access to the patient oriented data so that the unauthorized access can be prohibited.

Mackey et al. [45] proposes a blockchain-based system for fraud detection in healthcare insurance, achieved by using a smart contract that employs cryptographic encryption

mechanisms to secure medical data.

The use of blockchain technology in automated health care was first used by Kuo et al. [46]. The authors also developed a framework to reduce fraud and attacks in healthcare. The author used ModelChain to allow the nodes from authorised BC networks to participate in the network. This approach prevents 51% attack.

Innovative and cutting-edge solutions for medical data storage, transfer, processing, analysis, and categorization based on results are being developed using blockchain and AI. The IoT-based healthcare business is being revolutionized by blockchain technology, which improves efficiency, access control, technological advancement, privacy protection, and security of operational data. To address the issue of data security in smart cities, Rajawat et al. [47] suggested an architecture based on artificial intelligence (AI) and blockchain.

Schinkus et al. in [48] proposed a blockchain based application system to provide a secure platform for hospital related payment. The application has been developed on the bitcoin platform. Patient, insurance companies, and the hospital are three different kinds of node present in this dApp. All of three nodes use their network provided private key to register in the application. For payment purposes the public key of the receiver is used. After the payment, the blockchain network account is updated.

Singh et al. [49] proposed a blockchain based patient centric application system to secure the patient-oriented data. This is achieved by using a smart contract-based identity management system written in java script. The proposed system has been implemented in Hyperledger framework.

In [50] Angelis et al. developed a patient-centric data sharing system based on the ripple environment. The machine learning algorithms had been implemented in order to detect anomaly during the message passing. Once the attackers interpret the communication then the digital signature will be changed automatically which must be checked at the receiver side in order to find out the integrity of the shared message.

Angraal et al. [51] developed a dApp for insurance data on an open-chain environment based on the hash function, digital signature, and smart contract. When the patient goes for an insurance claim, the smart contract will be automatically invoked to verify their identity. The smart contract uses the unique id provided by the blockchain to verify the identity of the patient claiming insurance. The digital is used by the insurance company and the patient for approving the insurance claiming process. The hash function is used to create and manage the blockchain address of the participating node.

Dagher et al [52] had proposed a payment portal based on the bitcoin environment where the node chooses a particular vendor who has successfully submitted the nonce in a minimal time period. Two different blockchain networks connected with a blockchain bridge are used to register the patient and insurance company. At the time of registration with a particular insurance company, the patient broadcasts a nonce to the network of

insurance. The company submitting the nonce with a minimum time stamp will be chosen. During the insurance claiming process, the metamask payment wallet will be used to transfer the ether from one account to another.

Dhagarra et al. [53] developed a DApp based on the digital ledger technology (DLT). Once the patient had been verified by one doctor then the records will be stored in the DLT which is accessible for every participant node present in the network. The main limitation presented in this work is that the patient is unable to choose a particular participant for accessing the stored data. Every patient record is stored in the DLT with respect to its unique network. Table 2.3 shows the summary of the reported literatures.

2.1.4 Supply Chain Management (SCM)

Combining blockchain technology, IoT, and machine learning, the proposed system in [54] effectively addresses three issues of vaccination supply chain: quality, need, and user trust. The immutability of blockchain increases trust between participants. The efficacy of vaccines can be ensured by real-time IoT monitoring. Machine learning algorithms are used by vaccine producers in order to anticipate market demand and examine consumer feedback in order to improve product quality.

Agrawal et al. [55] proposed a blockchain-based system using forward and backward supply chain mathematical models, to reduce the time and money spent on transporting drugs from the manufacturer to the consumer. In particular, the forward chain concept promotes dependable, expedited transfer of pharmaceuticals from producer to consumer. Reducing the manufacturer's additional time and expense in pursuing a recall of the faulty medicine is a primary goal of the backward supply chain model.

Traceability is an essential component in the management of agricultural supply chains, and it plays an important role in ensuring the safety of food, which in turn increases consumer happiness and loyalty. Ehsan et al. [56] reported a traceability model that is based on blockchain technology and is completely decentralized. This model assures the system's integrity and transparency. The majority of the drawbacks of the conventional supply chain were eradicated by the implementation of this new model.

Bhat et al [57] proposed a blockchain-based supply chain system in coordination with the IoT system to avoid the limitation present in the conventional supply chain system. The major limitation present in the conventional security system is the security issue. The proposed system uses blockchain technology to introduce immutability and decentralization features to deal with the issue. A completely decentralized, blockchain-based traceability system for Agri-Food supply chain management was presented by Caro et al [58]. This solution is able to incorporate Internet of Things devices that produce and consume digital data throughout the chain in a smooth manner.

Shahid et al. [59] proposed a comprehensive solution for an agriculture and food supply

Table 2.3: Summary of applications of blockchain in Healthcare

Reference	Application System	Key Functionality
Huang et al. [37]	Healthcare data accessibility	a blockchain based healthcare application to detect the data modification in the stored data.
Zhuang et al. [38]	Data storage management	A healthcare application in British Columbia that relies on smart contracts to govern data storage and access.
Guo et al. [39]	Healthcare data accessibility	a blockchain based model to control the access to the health-related data.
Kim et al. [40]	Data security	A blockchain-based, trustworthy solution to store EHRs of patients in the cloud
Chenthara et al. [41]	EHR privacy	A blockchain-based architecture based on British Columbia for safeguarding EHR privacy.
Chen et al. [42]	Data Security	a BC-based EHR encryption system that enables users to search the information through a specific set of indexes stored in the blockchain.
Karmakar et al. [43]	Healthcare Insurance management	A blockchain-based solution to secure the healthcare based insurance system
Saldamli et al. [44]	Healthcare Insurance Security	A blockchain based application for securing the healthcare insurance system
Mackey et al. [45]	Healthcare Insurance fraud detection	A blockchain based system for healthcare insurance fraud detection.
Kuo et al. [46]	Reducing healthcare fraud and attacks	A model that mitigates the threat of a 51% attack
Rajawat et al. [47]	Healthcare data security	To address the issue of data security in smart city based healthcare system.
Schinkus et al. in [48]	Hospital bill payment	A blockchain based application system to provide a secure platform for paying the hospital related payment.
Singh et al. [49]	Securing the patient-oriented data	A blockchain based patient centric application system to secure the patient-oriented data.
Angelis et al. [50]	Data sharing system	A patient-centric data sharing system based on the ripple environment.
Angraal et al. [51]	Insurance management system	A dApp to deal with the insurance.
Dagher et al [52]	Healthcare payment portal	A payment portal based on the bitcoin environment.
Dhagarra et al. [53]	Data storage	A dApp for secure storage of the patient-centric data.

chain that is based on blockchain technology. It does this by using the most important aspects of blockchain technology and smart contracts, which are then implemented on the Ethereum blockchain network. Although the blockchain makes data and records in the network immutable, it is not yet capable of solving some of the most significant issues that arise in the management of supply chains. These issues include the credibility of the organizations that are participating, the accountability of the trading process, and the traceability of the goods. As a result of this, the evolved supply chain requires a dependable system that can provide traceability and a trusted delivery mechanism. All of the transactions are recorded on the blockchain, which eventually uploads the data to the IPFS scheme, according to the scheme that has been presented. A solution that is effective, safe, and dependable may be ensured by a storage system that provides a hash of the information that is saved on the blockchain.

Assigning unique digital IDs to food goods on the blockchain would allow for their development conditions, batch numbers, and expiration dates to be tracked along the supply chain. The concept developed by Ahmed et al. [60] aims to reduce food waste, educate consumers about the impact of their eating habits on the environment, and direct the redistribution of edible food surpluses to those in need. The public and immutable registry of goods and transactions might help identify the origin of foodborne diseases and reduce the likelihood of fraud. Additionally, blockchain will encourage the exchange of on-farm data as digital technologies become more prevalent for overseeing farms. Table 2.4 shows the summary of the reported literature.

Table 2.4: Summary of applications of blockchain in Supply Chain Management

Reference	Application System	Key Functionality
Hu et al. [54]	Healthcare SCM	The proposed blockchain-based SCM effectively addresses three issues plaguing the vaccination supply chain: quality, need, and user trust.
Agrawal et al. [55]	Healthcare SCM	A blockchain-based system to reduce the time and money spent on transporting drugs from the manufacturer to the consumer.
Ehsan et al. [56]	Agricultural-based SCM	It plays an important role in ensuring the safety of food, which in turn increases consumer happiness and loyalty.
Bhat et al. [57]	Agricultural-based SCM	A blockchain-based supply chain system in coordination with the IoT system to ensure food security.
Caro et al. [58]	Agricultural-based SCM	A blockchain-based traceability system for Agri-Food supply chain management.
Shahid et al. [59]	Agricultural-based SCM	A comprehensive solution for an agriculture and food supply chain.
Ahmed et al. [60]	Food SCM	A model to reduce food waste, educate consumers about the impact of their eating habits on the environment, and direct the redistribution of edible food surpluses to those in need.

2.1.5 Financial Sector

Singh et al. [61] proposed a blockchain-based finance system for lightweight devices, especially for the healthcare sector. The system can also be applied to other sectors. The main benefit of this developed application system is the implementation of a zero-knowledge proof mechanism which enables the system to work more efficiently. In this model, only a few milliseconds are required to validate the transaction. The app also reduces the communication cost between the nodes.

Son et al. [62] proposed a blockchain-based financial system for making a quick financial settlement. In the conventional system, the user may cause a delay in settling the payment related to the bank loan. Late approval or signature can cause a delay in the

payment. However, the proposed system deploys a smart contract-based system to ensure that the settlement is done on the appropriate date.

Saxena et al. [63] created a blockchain-based financial system inspired by the notion of Bitcoin. This is a real-world application that is fully automated and characterized by complete openness, trustworthiness, and independence from a central authority. It is entirely transparent since the network is in the public domain and decentralized. The transactions between peers, or nodes in the network are recorded in the DLT. PoW consensus mechanism is used to select the CA or validator for a transaction. However, the use of PoW increases the network overhead and makes it difficult for the lightweight devices to become the validator or CA.

Chen et al. [64] proposed a financial system, for making inter-organizational payments. The developed system provides a streamlined application for making trust-less transactions. Case studies from two Eastern banks demonstrate how the technological capabilities of blockchain can help reduce ambiguous actions and increase trust between businesses.

Chuah et al. [65] proposed a blockchain-based system for anticipating the money-laundering system. Before starting a transaction the proposed dApp invokes a smart contract for verifying the identity of the initiating node. The verified nodes are only allowed to execute transactions using the private key of the receiver.

Jiang et al. [66] suggested a blockchain-based payment model for small and medium scale industries. It uses direct and indirect trust via the use of intuitionistic fuzzy set theory to determine the validator for the transaction. Trust aggregation is the key feature behind the proposed model. The trust is calculated in two different ways such as direct and indirect trust. The proposed model deploys a trust transitivity model with a smart contract to calculate the trust for every transaction.

Zhang et al. [67] proposed a blockchain-based payment system for universities. The developed system merges all financial systems in a single application. Every financial transaction is added to the DLT of the blockchain network in order to achieve tamper-proof transaction.

With the advent of blockchain technology, the banking system is now a day accelerating its movement toward the digital currency system. Zhang et al. [68] analyzed the functional and non-functional requirements of a typical banking system for transitioning from a legacy system to a blockchain-based banking system. The detailed analysis shows that a private blockchain environment is more preferable compared to a public blockchain for this kind of applications.

A Consortium Blockchain-based overseas money transfer mechanism was proposed by Patil et al. [69] which achieved swifter operations, security, and transparency. The Hyperledger Fabric Blockchain infrastructure with web-based user interface was implemented for the procedure of transferring funds internationally.

Ranjan et al. [70] proposed a blockchain-based application system to support the government finance system. The proposed system can reduce corruption in transferring fund to the vendor's account. In addition, the proposed system provides a traceable environment to track the fund utilization. This system provides clear insight into how and what percentage of funds are being used.

The logistics industry could benefit from the blockchain-based application implemented by Fu et al.'s [71]. It uses a smart contract technology which is efficient and interpretable. In addition, it also uses a consensus algorithm, to achieve automatic control of privacy information flow. It streamlines the process of securing the financial system of logistics companies while protecting the confidentiality of their customers' personal information.

Kabra et al. [72] proposed a blockchain-based finance system named as $MudraChain$ for automating the cheque clearance system in banking. Traditionally, the cheque clearance process is executed by a truncation system where all possible activities are cleared manually. The proposed system uses blockchain technology to automate this clearance process. This process includes the blockchain-based authentication system, signing the cheque digitally, and a two-way authentication scheme to remove the traditional truncation system.

Patel et al. [73] proposed a blockchain-based credit recommender system using an AI-based approach. The main objective of the developed system is to remove the presence of third-party agencies. This system creates a streamlined process between the lander and borrower to enable a smart contract-oriented automatic loan disbursal system after fulfilling the criteria.

Table 2.5: Summary of applications of blockchain in Financial Sector

Reference	Application System	Key Functionality
Singh et al. [61]	Finance system for healthcare	A blockchain-based finance system for lightweight devices
Son et al. [62]	Loan Settlement	A blockchain-based financial system for making a quick financial settlement.
Saxena et al. [63]	Autopay system	A blockchain-based autopay financial system inspired by the notion of Bitcoin.
Chen et al. [64]	Payment System	A financial system, for making inter-organizational payments.
Jiang et al. [66]	Payment System	A blockchain-based payment system for universities.
Zhang et al. [67]	Payment System	A BC-based EHR encryption system that enables users to search the information through a specific set of indexes stored in the blockchain.
Zhang et al. [68]	Banking System	A blockchain-based solution for switching legacy banking system to blockchain based banking system
Patil et al. [69]	Payment System	Blockchain-based overseas money transfer mechanism
Ranjan et al. [70]	Payment Portal with web-based UI	A blockchain-based application system to support the government finance system.
Fu et al. [71]	Finance system of logistics companies	A finance system of logistics companies while protecting the confidentiality of their customer's personal information.
Kabra et al. [72]	Automating the cheque clearance system	A blockchain-based finance system MudraChain for automating the cheque clearance system in banking.
Patel et al. [73]	Credit recommender system	A blockchain-based credit recommender system using an AI-based approach.

2.2 Applications of blockchain in security

This section shows a detailed study of different blockchain-based attacks 2.2.1 and different blockchain-based PKIs 2.2.2 for providing security solutions to different attacks.

2.2.1 Study on Blockchain-based attack

Blockchain attacks are classified into four different categories such as network attacks, transaction attacks, consensus attacks, and wallet-based attacks. These attacks have the potential to affect the pervasive features of blockchain technology.

Kausar et al [74] reported various types of attacks on the blockchain network. According to the study the attacks can be classified into different types such as Finney attacks, Flooding attacks, and Block with-holding attacks (BWH). The Finney attack is a type of selfish mining attack. The Finney attack occurs when a transaction is pre-mined into a block and an identical transaction is broadcasted to the network which makes initial transaction invalid. The A flooding attack is the process of increasing network overhead by generating a large number of fake transactions. In the BWH attack, the attacker joins the pool of miners. Unlike the genuine miner, the attacker tries to submit the partial PoW ($FPoW$) instead of the full PoW ($FPoW$) to get paid. Among all attacks studied, the research shows that flooding attacks and block withholding attacks have the most significant impact on the blockchain network.

Chaganti et al. [75] reported three main types of attacks on the blockchain network: double spending attacks, DDoS attacks, and DoS attacks. Among these three, double spending attacks occur in the Bitcoin network using the PoW consensus mechanism.

Saad et al. [76] mainly focused on the cryptographical attack or Man In the Middle ($MITM$) attack on the network transaction in which the unethical mining of the blockchain network occurs. This kind of attack generally, occurs to the blockchain network such as the Bitcoin platform using the PoW consensus mechanism. This attack tampers the transactional data in the network.

51% attack is another impactful attack in blockchain network reported in [77]. This attack aims to slow down the entire blockchain network by occupying more than 51% of the participants. The attacker with this kind of attack focuses on becoming the validator for every possible transaction in the network resulting which the entire network becoming malicious.

Sengupta et al. [78] reported the Sybil attack as another blockchain-based attack. In this attack, the attacker tries to attack a single node and initiates Sybil or fake identities to perform different malicious transactions. However, the blockchain network does suffer more from this kind of attack as the entire network does not depend on any single node to perform the transaction.

The Eclipse attack reported in [79] is the extended version of the Sybil attack where the attacker tries to deploy multiple bot nodes for active participant nodes. Resulting this attack the network may suffer from multiple malicious transactions which directly affect the network performance. Fu et al. in [80] reported another kind of possible attack in blockchain named Time jacking attack. It is an attack that uses a potential flaw in how Bitcoin handles timestamps. A node may be time jacked if an attacker manipulates its network time counter. An attacker may do this by populating the network with several fraudulent nodes which play a vital role in executing the transaction whose timestamps are wrong.

Routing and DAO are one of the most powerful attacks present in blockchain networks [81]. Routing attack aims to manipulate the transactional data before it is delivered to its corresponding peer. In addition, this also aims to increase the block propagation delay through which the network performance decreases severely. In addition, this kind of attack may create different partitions within the network and forces a malicious node as the linking peer in between the partitions. A decentralized Autonomous Organization (DAO) is responsible to create smart contracts for different organizations with the blockchain network. A small change in the smart contract may cause an organization to become malicious. In a DAO attack, the attacker tries to compromise the DAO to produce compromised smart contracts.

Race attacks [82] and Finney attacks [83] are two other types of attacks that focus on the wallet of the peer. An attacker carries out the race attack by generating two conflicting transactions. The victim receives the first transaction and accepts the transaction by paying the mentioned amount without waiting for confirmation from the network. The attacker simultaneously broadcasts a conflicting transaction to the network in which the identical amount of bitcoin is returned to the attacker, rendering the original transaction invalid.

The selfish-mining attack [84] is another type of attack on the Blockchain network. This attack mainly focuses on cryptocurrencies. A malicious node enhances the share of the reward by simply avoiding the broadcast of mined blocks into the network. Then, after some time intervals, several blocks are released simultaneously resulting which other nodes or miners losing their blocks. Successful selfish mining is possible with any cryptocurrency.

Fork After Withholding (FAW) [85] is a variant of a selfish mining attack that benefits attackers more than the original protocol. In a FAW attack, a rogue miner will temporarily withhold a winning or completed block before discarding it or releasing it to cause a fork. This attack results in the mining pool losing the rewarded bitcoins.

Time Delay attack [86] is another prominent attack on the blockchain network. In this kind of attack, the attacker tries to identify all possible peers of the network. Then, the attacker tries to increase the computational load on the network by overloading all active participants in the network. Increasing the computational overhead also increases

the delay in transaction, block creation, and block propagation.

Table 2.6 provides insight into different attacks on the Blockchain network. It can be observed that the attacks including the Flooding attack, BWH attack, DoS, DDoS, MITM, 51%, Sybil attack, Eclipse attack, and Time Delay attack put more impact on the Blockchain network performance.

2.2.2 Study on Blockchain-based PKIs

Garba et al. [87] developed a PKI for IoT-enabled networks. The main objective of the developed PKI is to verify the certificates in an IoT-based network. In addition, the developed PKI focuses on the generation of light weigh certificates that can be easily verified by the *Registration Authority*. Each node of the network is responsible for the creation of its own certificate. Then the generated certificates are forwarded to the Local Registration Authority (LRA) for verification. Once the certificate is verified, the node is allowed to initiate the communication.

Zhai et al. [88] developed a Blockchain-based PKI solution BPKI to identify the malicious activity of the CA in the network. The developed PKI implements auditors which are responsible for regularly monitoring the activity of the selected CA. The main objective of the BPKI is to detect the domain name pre-emption attack to the CA. The author has used a double Blockchain structure to solve the scalability issue present in the Blockchain concept.

Wang et al. [89] proposed a DAG-based Blockchain solution for authenticating the identity of the IoT devices. Low-power DLT is used to provide a lightweight, scalable method for managing the identities of IoT devices. In addition, the proposed system aims to preserve access control to ensure the secure and reliable sharing of network data. In addition, the adopted DLT is used for registering, updating, revoking, and retrieving the identities of the connected devices of the network. The enormous quantities of data produced by IoT networks may be stored using the interplanetary file system (IPFS), which not only relieves pressure on IoT network storage systems but also eliminates the bottlenecks, delays, and other drawbacks associated with conventional cloud storage. In addition, the fog node is used to address the issue of insufficient computing resources of the IoT devices by providing a localized shared computing environment for performing any kind of complex task.

To address the scalability issue of the $BC-PKI$, Xu et al. presented ScalaCert [90] as a Blockchain-based PKI solution. A redactable blockchain was used to revoke certificates to accomplish this. Specifically, revocation information is recorded on the original certificate through the redactable blockchain. Further storage overhead is eliminated because of the elimination of heavy weight data structures like CRL. In addition, the proposed PKI implements a random Blockchain nodes check (RBNC) to verify the legitimacy of

all nodes involved. A new kind of attack, known as a deception of versions (DoV) attack, emerges with the introduction of redactable and consortium blockchains. To defend against it, RBNC methodology is used.

Another Blockchain-based PKI ETHERST has been developed in [91]. The main idea present behind the ETHERST is the rewarding and punishing mechanism applied to the CA of a transaction. ERC-20 token is introduced as a mechanism to achieve this. In Blockchain-based PKIs, there is a way to detect the malicious CA present in the network. However, there is no such method available to penalize that malicious CA. The developed ETHERST uses a smart contract to reduce 20 ETH from a CA Ethereum account if found malicious. At regular interval, the selected CAs are verified for any kind of malicious activity. If any CA is found to produce a malicious certificate, then the smart contract is invoked to deduct 20 ETH. After the deduction of the ETH that node will immediately be removed from the network.

Xie et al. [92] proposed a $BC-PKI$solution CR-BA with the main focus on certificate revocation in the network. During the certificate creation process, the CA introduces the revocation counter and a smart contract account in the certificate. When the certificate revocation request is received the revocation fingerprint is generated and broadcasted to the network. The broadcasted result is stored securely in the DLT. When a node from the network wants to verify the status of the certificate then, the CA calculates the revocation counter present in the certificate and generates the fingerprint. This generated fingerprint is compared in contrast to the stored figure print in DLT. The comparison result is sent to the requested user from the status of the certificate can be achieved.

Wang et al. [93] proposed a $BC-PKI$for improving the certificate revocation process. The author uses two different types of filters to store the revoked and validated certificates. Once the certificate is generated and validated, it will be immediately moved to one filter. The second filter is used to store the status of the revocation process and the list of certificates that are successfully revoked. The proposed PKI is implemented and evaluated over the Hyperledger framework. The empirical analysis shows that the proposed PKI outperforms a few $BC-PKI$by optimizing the certificate revocation time by using the second filter.

Moussaoui et al. [94] proposed a $BC-PKI$to increase the security aspect of VANET. In VANET security and privacy are the two key factors which become the main challenging factor in this field. The present solutions for maintaining the security aspect is the use of pseudonyms. The pseudonym is a digital certificate that contains hidden information about the vehicle. The main three processes of the pseudonym are creation, issue, and revocation. In the proposed model two blockchain models are used. One is to do the pseudonym operations including the creation and issue. The second pseudonym is used for the revocation process to reduce the time required to revoke the certificate.

Obiri et al. [95] proposed a $BC-PKI$for IoT-based networks. $BC-PKI$uses the

CA to generate and validate the certificates. The processing time may increase if the CA receives multiple requests to generate the certificates. To optimize this time the proposed PKI enables the IoT devices or nodes to maintain their own private and public key pair. The public key is stored in a separate database with blockchain technology as the proof of concept. The operations including the node authentication are done in an off-chain manner without requiring the smart contract invocation.

Zhang et al. [96] proposed a $BC-PKI$named BPAF for authenticating Fog-based IoT devices. Conventional blockchain-based identity management cannot be applied directly to the blockchain network due to the IoT device resource constraint. The proposed PKI introduces a lightweight authentication scheme for IoT-Fog-enabled networks. The participating IoT devices need not to store the DLT. Instead, the blockchain will have some full nodes with higher computational and storage capacity used to store the DLT. In addition, the full node also stores the list of public keys and a certificate revocation list. Fog nodes are considered as thin nodes of the blockchain network. The IoT device requests the Fog-node for verification and revocation of the digital certificate. The fog node then forwards the query to the full node for verification. The proposed PKI uses the BFT consensus mechanism to make the network more robust.

Garba et al. [97] proposed a novel approach for a digital certificate and domain authentication in a blockchain network especially for thin devices. Initially, the PKI stores the set of trusted CA with their corresponding domain name. Whenever the recorded CA is requested to issue a certificate for a transaction, then the network participants first verify the domain for trustworthiness. If found trusted, then the selected CA is allowed to issue the certificate for the transaction. For validating the domain the light smart contracts are used, so that the thin devices can also take part in the verification process.

Sermpinis et al. [98] proposed DeTRACT as a decentralized $BC-PKI$for certificate verification. The proposed system aims to validate website-based client-server communications. The nodes create the self-signed certificates and store them either in Bitcoin or Ethereum blockchain platform. The developed PKI uses the self-signed certificates to validate the domains included in the website-based communications. The status of the verification process is stored in a DLT to inform all other participant nodes present in the network, thus making the verification process more transparent. The proposed $BC-PKI$is implemented in two blockchain platform such as Bitcoin and Ethereum.

BlockCAM [143] is another BC-PKI with an objective of cross-domain verification. It employs a set of nodes to verify the certificate in different domains. The hash value of the verified certificate is recorded in the blocks. This hash value is used to retrieve the certificate information during the revocation process. The CA computes the hash value of the requested certificate and compares it with the stored value in the block. If matching is found then only the CA revokes the certificate.

BC-Trust [144] is another BC-PKI dedicated to the PGP server. The main objective

of this PKI is to minimize the certificate revocation time and to prevent the $MITM$ attacks on the certificate. In addition, it also provides a granular access control to the certificate owner to modify the certificate at any time.

TTA-SC [147] is one more BC-PKI based on a web-of-trust approach. The proposed PKI provides a dynamic trust evaluation method for the issues . For CA selection the consensus algorithm is used to choose a node with high trust. Once the node is declared as the untrusted then its issued certificates are also revoked from the DLT and the active participant nodes of the nodes are informed regarding this.

CERT-CHAIN [148] is a BC-PKI developed to enhance the security aspects of SSL/TLS connections. The PKI used a log to maintain the status of the CA. At regular interval this status is broadcasted to the network to inform all the participating nodes about the malicious nodes. The PKI deploys a dedicated node to handle this log. Dependability Rank-based (DRB) consensus mechanism is used to select the node as CA performing highest number of transactions in the network.

Instant Kerma PKI (IKP)[151] is another blockchain based PKI. The main focus of IKP is to identify the malicious CA. A separate list of verified and non-malicious lists of CA is maintained in the DLT. It also provides incentives to the group of nodes helping in identifying the malicious CA. The nodes of the network have to choose the validator for the transaction from the list maintained in the DLT.

2.3 Analysis

The centralized application suffers from various limitations such as single point failure and security issues . The reported literatures show that the features of blockchain including decentralization, immutabilty, and security helps in avoiding the limitations of the centralized applications. The critical analysis of the reported literature can be summarized as follows:

- Among all application domains of the blockchain the healthcare sector and financial sector requires privacy while doing the transaction. The users in these two sectors require granular access to the data to avoid public access. As a proof of concept the healthcare sector can be considered as a perferable area for implementaing the blockchain technology.

- Most of the existing literature focuses on MITM, DoS, and DDoS attacks while keeping all other types of blockchain-based attacks unexplorable.

- None of the existing literature focuses on optimizing the network overhead caused due to the CA selection process. The search space for a CA selection process in-

creases with the increase in the network traffic resulting. This may result in increasing the time and overhead leading to various delay including the block propagation delay.

Table 2.6: Different attacks to the Blockchain network

Attack	Target Area	Characteristics	Impact on Blockchain
Finney attack	Wallet	A transaction is pre-mined into a block and an identical transaction is broadcasted to the network resulting which the initial transaction becoming invalid	Low
Flooding attack	Network	The process of increasing the network overhead by increasing the number of fake transactions	High
Block withholding (BWH) attack	Wallet	Submitting PPoW instead of FPoW to get paid	High
Double Spending attack	Wallet	Same transaction to multiple nodes sent simultaneously	Low
DoS	Network	Transaction flooding to a particular node of the network to decrease its efficiency	High
DDoS	Network	Transaction flooding to a group of nodes to decrease its efficiency of the network	High
MITM	Transaction	The unethical modification in the Blockchain data	High
51%	Network & Transaction	To slow down the entire blockchain network by occupying more than 51% of the participants	High
Sybil attack	Network	To attack a single node and initiates Sybil or fake identities to perform different malicious transactions	High
Eclipse attack	Network	To deploy multiple bot nodes for active participant nodes to perform malicious transactions	High
Time Jacking attack	Network	To populate the network with several fraudulent nodes which play a vital role in executing the transaction with wrong time stamp	Moderate
Routing attack	Transaction	To manipulate the transactional data before it is delivered to its corresponding peer	High
DAO attack	Consensus Method	To produce compromised smart contracts	Moderate
Race attack	Wallet	To broadcast a competing transaction to the network in which the identical amount of bitcoin is returned to the attacker, rendering the original transaction invalid	Moderate
Selfish mining attack	Wallet	A malicious node enhances the share of the reward by simply avoiding the broadcast of mined blocks into the network	Moderate
Fork After Withholding (FAW)	Wallet	A rogue miner will temporarily withhold a winning or completed block before discarding it or releasing it to cause a fork	Moderate
Time Delay attack	Network	To increase the computational load on the network to overload all active participants in the network to increase the delay in transaction, block creation, and block propagation	High

Table 2.7: BC-PKI solution sumary

Method	Attack Prevention							
BC-PKI	DoS	DDoS	MITM	51%	Injection	Routing	Eclipse	Network Overhead Optimization
Participant	DoS	DDoS	MITM	51%	Injection	Routing	Eclipse	Network Overhead Optimization
Signature	DoS	DDoS	MITM	51%	Injection	Routing	Eclipse	Network Overhead Optimization
Revocation	DoS	DDoS	MITM	51%	Injection	Routing	Eclipse	Network Overhead Optimization
Smart Contract	DoS	DDoS	MITM	51%	Injection	Routing	Eclipse	Network Overhead Optimization
Smart Contract	DoS	DDoS	MITM	51%	Injection	Routing	Eclipse	Network Overhead Optimization
Smart Contract	DoS	DDoS	MITM	51%	Injection	Routing	Eclipse	Network Overhead Optimization
Smart Contract	DoS	DDoS	MITM	51%	Injection	Routing	Eclipse	Network Overhead Optimization
Smart Contract	DoS	DDoS	MITM	51%	Injection	Routing	Eclipse	Network Overhead Optimization
Smart Contract	DoS	DDoS	MITM	51%	Injection	Routing	Eclipse	Network Overhead Optimization
Smart Contract	DoS	DDoS	MITM	51%	Injection	Routing	Eclipse	Network Overhead Optimization
Smart Contract	DoS	DDoS	MITM	51%	Injection	Routing	Eclipse	Network Overhead Optimization
Smart	DoS	DDoS	MITM	51%	Injection	Routing	Eclipse	Network

Chapter 3

A blockchain-based Decentralized Application for Health Care System

Blockchain technology is currently playing a significant role in providing a secure and effective means to share information in a variety of domains, such as financial sector, supply chain management (SCM), IoT, and the field of health care systems (HCS). The interoperability and security of HCS allow patients and vendors to communicate information seamlessly. The absence of these properties increases patient's difficulties to access his or her own health information. The adoption of blockchain technology in HCS eliminates this disadvantage, allowing the HCS to become more effective and efficient. These potential benefits provide a foundation for blockchain technology to be used in various aspects of HCS, such as maintaining the patient electronic health record (EHR) and electronic medical records (EMR) for various medical devices, billing, telemedicine systems, and so on. In recent years, decentralized applications or dApps have been rapidly emerging as a promising research topic and being adopted by various fields such as banking, medical and business, etc. The dApps are nothing but digital applications which run on a peer-to-peer network outside the purview and control of a single controlling body. This chapter focuses on developing a decentralized application EHR for storing and sharing medical data between a patient and a doctor. This is a very basic implementation of a blockchain-based EHR application that serves as an introduction to the Ethereum platform. However, this chapter does not examine any relevant performance parameters.

3.1 Introduction

In a peer-to-peer network, blockchain provides a safe and advanced network for executing and exchanging information between multiple nodes. According to Gartner, blockchain is

one of the top ten most important innovation trends for 2018 [99]. It is stated in [100][1] that using a public blockchain can reduce the need for trustworthy nodes for exchanging information. The ongoing transactions are validated by that node alone if a trustworthy node is deployed. When data is exchanged in a blockchain network, three primary components are present: Blocks, Nodes, and Miners. Miners create new blocks in the network, which is referred to as mining [101]. The preceding block's hash value must be remembered and referenced when generating the new block. Along with creating new blocks, miners also contribute to the solution of the NONCE in order to become the authority for certifying a transaction [102]. When a block is successfully mined, all nodes in the network agree on a value, and the miner is rewarded financially. In blockchain technology, the node is the most critical element which uses DLT in the network for sharing the information [103]. The nodes have a copy of the blockchain in DLT, and any mining that takes place within the network must be approved by the network as a whole. Blockchain transparency allows the patients to view and examine the corresponding EHRs stored in the network [104].

Since Satoshi Nakamoto introduced blockchain technology in the form of bitcoin, it has developed quickly and attracted the attention of numerous academic and commercial researchers. [105, 106]. Blockchain technology is a decentralized system that is deployed in a peer-to-peer network to store transactional information, also known as blocks, in a public database called a distributed ledger that is accessible to any active network participant [107]. Due to features such as decentralization, immutability, security, and transparency, blockchain technology is becoming the most promising and prominent technology advent for internet-based communication [108].

Secure and scalable data sharing is essential for the healthcare decision-making system. Traditional clinical data initiatives, on the other hand, are typically fragmented, impeding effective information flow thus preventing a patient from making sensible treatment decisions [109]. Blockchain technology plays a vital role in providing a secure platform for storing and sharing medical records between a patient and a doctor. Implementing dApp has its own benefits and challenges with respect to blockchain technology [110, 111]. The benefits and challenges of the healthcare system based on dApp are addressed in Table 3.1 and 3.2 respectively.

3.2 Background Study

This section shows an overview of decentralized application, along with MetaMask and Ethereum blockchain platform which are the key component of this developed work.

[1]The outcomes of this chapter was published in "The International Journal of Information Systems and Supply Chain Management (IJISSCM)"[J1]"

Table 3.1: Advantages of blockchain technology in the healthcare system

Functionality	Benefits in healthcare
Network Structure	The peer network structure provides a secure infrastructure
Cryptography Mechanism	Enables the system to prevent the unauthorized access
Distributed Ledger	Secure access control
NONCE	Acts as the authorization process in choosing the validator of a transaction
Smart Contracts	Helps in increasing transparency and transaction execution by automating the process
WoT	Dependable election mechanism for choosing a validator for one transaction
Permissioned Transaction	Since any sort of data alteration requires authorization from all parties, the degree of interference with stored data is decreased

Table 3.2: Limitation of blockchain in the healthcare system

Functioality	Limitations in healthcare
Storage Limitation	A considerable amount of storage is required to store a large amount of hospital and patient-centric data, which is an extremely challenging task [112]
Dynamic Records	Healthcare data is constantly changing. Data changes every second and must be saved in blockchain blocks on a regular basis. The alteration procedure takes a long time because authorization from each participant is required, which adds to the time complexity [113]
Network scalability	The blockchain's decentralized framework makes it difficult to add more healthcare systems to the existing blockchain
Vendor Interest	There are various systems that have no interest in sharing EHR as they prefer to follow the legacy system for maintaining those
Shifting the traditional system to dApp	Doctors are following the conventional way while writing prescriptions for their patients and show little interest in EMRs. The transition from the legacy system to the BCT is extremely difficult [114]

3.2.1 Dcentralized Application:

Decentralized applications or dApps refer to any application that may be executed on the blockchain platform. It is developed with an objective to be executed in a distributed environment instead of on a single system. The dApps is executed outside the control

and scope of a centralized controller. With the advent of internet based communications the dApps are emerging as a new category of applications using Blockchain and smart contracts. Smart contracts on a blockchain store the essential information and activities. The dApps interact with smart contracts and provides services depending on transactions using the contract requests.In order to utilise a smart contract, users still have to execute the programmes on their local machines. The inability of existing blockchain technology to execute at a level sufficient for many uses is a major factor. This raises concerns about application upkeep and operational safety. It's possible, for instance, that there are local fraudulent behaviours that are intentionally concealed from the public assessment.

However, at present, dApps may only use smart contracts for the most fundamental data and features that should be immutable. In order to utilise a smart contract, users still have to execute the programmes on their local machines. The inability of existing blockchain technology to execute at a level sufficient for many uses is a major factor. This raises concerns about application upkeep and operational safety. It's possible, for instance, that there are local fraudulent behaviours that are intentionally concealed from the public assessment.

A decentralised application (dApp) hosted entirely on a peer-to-peer blockchain system should be considered the pinnacle of blockchain technology. Once a dApp is released, it won't need any more oversight or upkeep from its developers. In other terms, a DAO is formed so that the dApps can run without any human involvement. A decentralised autonomous organisation (DAO) is an organization whose operations are governed by specific regulations stored in smart contracts that are executed on the blockchain. The cost and profit of a DAO are distributed equally among all peers automatically by recording all transactions in the blocks transparently. The most well-known blockchain technology, Bitcoin, is itself a DAO. The features of the dApps are as follows

- **Open Source:**The decentralized and immutable nature of blockchain necessitates that dApps provide their source code for independent third party verification.
- **No Central Authority:**The use of decentralization avoids the requirement of central authority to control the application. Thus increasing the robustness of the dApp.
- **Cryptocurrency Support:** Using this the peers of the network can send and receive the native cryptocurrency by using the corresponding public key.

3.2.2 Metamask:

Metamask is an extension for web browsers and mobile application. It enables the users to safely connect to decentralised services. The decentralized services includes managing

account identifiers such as public keys , broadcasting the transaction. In addition to the above said feature Metamask is mainly used for sending and receiving the Ethereum-based cryptocurrency.

3.2.3 Remix IDE:

An integrated development environment (IDE) is essential for creating smart contracts and managing their whole lifespan, including compilation, testing and deployment. There are a large number of options available for developing the smart contracts for blockchain based applications. One such web-based IDE is Remix IDE. Remix includes a comprehensive set of tools for developing, deploying, and managing the smart contracts.

3.3 Need of blockchain based healthcare system

For developing a healthcare-based application various factors need to be considered. In contrast to other techniques, blockchain technology is widely used in developing a HIT application due to its decentralized characteristics. For developing HIT-based applications the requirements need to be acknowledged [115]. Various application systems use different kinds of use cases with different technical requirements [116]. In this current section, those issues are addressed along with various techniques to deal with those concerns. When developing a HIT-based application the privacy is a primary attribute that needs to be focused [49]. According to the typical blockchain concept, not all transactions in health care should be made. There are regulatory and legal requirements that must be observed when handling healthcare data. As a result, any blockchain architecture that is utilized to build healthcare apps should have a thorough set of privacy safeguards [117, 118].

Security is closely tied to privacy. To prevent all types of data theft, HIT systems must be established and developed. In HIT every actor should be easily recognizable, as should their behaviors. Security criteria for healthcare apps, like privacy, are enforced by rules and must be followed [104]. Blockchain technology should provide a powerful authentication and access control technique for regulating the participating node and the data. The transaction throughput is another factor to consider when selecting a technology stack for designing healthcare applications. In some circumstances, such as remote patient monitoring (RPM) systems, healthcare applications must be able to grow in terms of speed and transaction throughput [119]. The number of nodes that can participate in the consensus mechanism determines the transaction throughput or scalability of blockchain frameworks [120]. Consensus is concerned with how the blockchain network's transactions are processed and are discussed in more detail in the next section. They must first be validated before they can be regarded as valid transactions [121]. This number could be as high as all of the network's nodes or as low as a single network node. As a result, the

various blockchain frameworks have different techniques for achieving consensus, which are important concerns [122, 123].

3.4 Related Work

In [124] Azaria al. represents the problems faced during medical record sharing and also describes the solution based on blockchain technology to handle this. The developed work was based on the key agreement protocol to solve the issues of sharing the medical records. The dApp is was develop on a public blockchain with EMR as the key factor. In [125] Zhou et al. proposed a blockchain-based dApp for securing the information exchange between the different users. The application was developed on Ethereum environment with cryptography, nonce as the key functionality. Latif et al. [126] had developed supply chain management (scm) for dealing with the transaction history of the patient. Every transaction on the developed scm must be digitally signed by the user.

In [50] the author had developed a patient-centric data sharing system based on the ripple environment. The machine learning algorithms had been implemented in order to detect the anomaly during the message passing. Once the attackers interpret the communication then the digital signature will be changed automatically which must be checked at the receiver side in order to find out the integrity of the shared message. In [51] the authors developed a dApp on an open-chain environment based on the hash function, digital signature, and smart contract to deal with the insurance. The patient claiming for insurance should be verified in terms of digital signature with the help of a smart contract. Whenever the patient will go for an insurance claim then the smart contract will be invoked automatically to verify the identity.

Dagher et al in [52] had proposed a payment portal based on the bitcoin environment. The node will choose the particular vendor who has successfully submitted the nonce in a minimal time period. Upon getting the result the patient will deal with that particular insurer for insurance claiming and the metamask payment wallet will be used to transfer the ether from one account to another. In [53] the author had tried to develop a dApp based on the digital ledger technology (DLT). Once the patient had been verified by one doctor then the records will be stored in the used DLT which is accessible for every participant node present in the network. The main limitation present in this work is that the patient is unable to choose some particular participant for accessing the stored data. Every patient record is stored in the DLT with respect to its unique network. Dimitrov et al [127] had developed a ripple-based SCM for sharing the patient record in which the uid will be a key attribute for sharing the information. Every patient and doctor is a part of the network thus having unique ids. Every patient has to store the diagnosis report in the provided public ledger from where the doctor can have access. The limitation present

in this developed work is that the patient can not pay the doctor by using the blockchain wallet.

3.5 Objective

The contribution of this research work is to share the medical data between the patients and doctors to provide an improvised decision system for the health care system. The objective of this work is summarized below.

- To develop a dApp using the blockchain system for the health care system.
- To provide a cost estimation method for calculating the overall cost for implementing the developed system in a real-time scenario.

3.6 Proposed Work

The purpose of this framework is to integrate blockchain technology for EHR first and then to enable secure electronic record storage for users of the proposed framework by creating granular access controls. Furthermore, by utilizing off-chain record storage, this framework overcomes the scalability issue that blockchain technology has in general. This provides the EHR system with the benefits of a scalable, secure, and integrated blockchain-based solution.

3.6.1 Methodology

The Healthcare EHR dApp is written in Javascript. This dApp provides an easy-to-use UI allowing the users to share and view the patient data. The developed system contains different modules such as the Registration and Login module, EHR uploading module, HER Accessing module. Figure 3.1 and 3.2 shows the block diagram and use case diagram of the proposed system. The developed application is deployed by using the Ganache EVM, Metamask, IPFS little server with a system having Windows OS, 256 Gb SSD, 1 TB HDD, and i5 8th generation processor with 2.4-2.6 GHz clock speed.

3.6.1.1 Registration Process

Every user in terms of patient and doctor needs to register themselves for using the dApp. Before registering in the application, the users need to join the Ethereum network so that each can have their own key pair. The registration process will be done by using the user details such as the name, age, the public key, and the activity for joining the application through the given UI. The attributes needed for registration are depicted in Figure 3.3. After successful registration, the user needs to log in by using the provided public key.

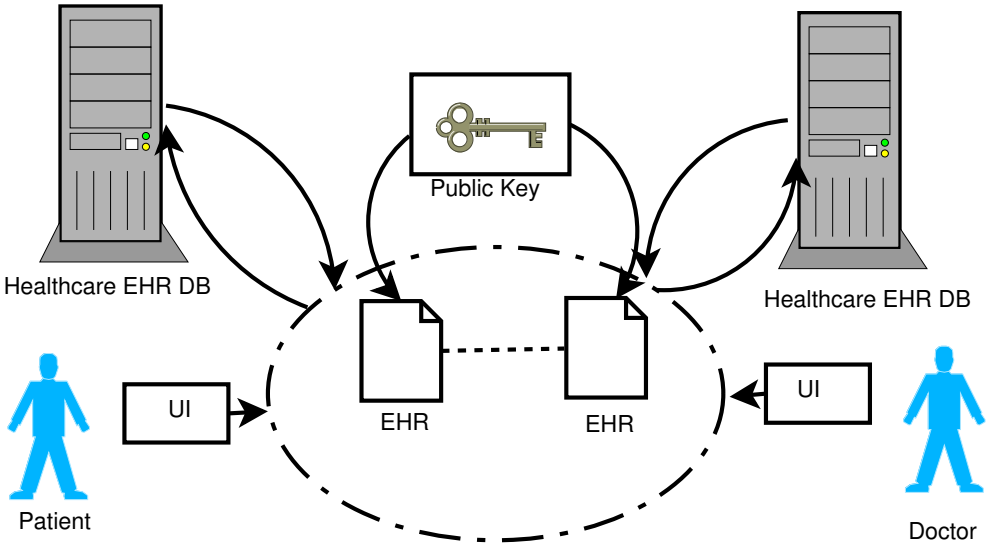

Figure 3.1: Block diagram for the proposed system

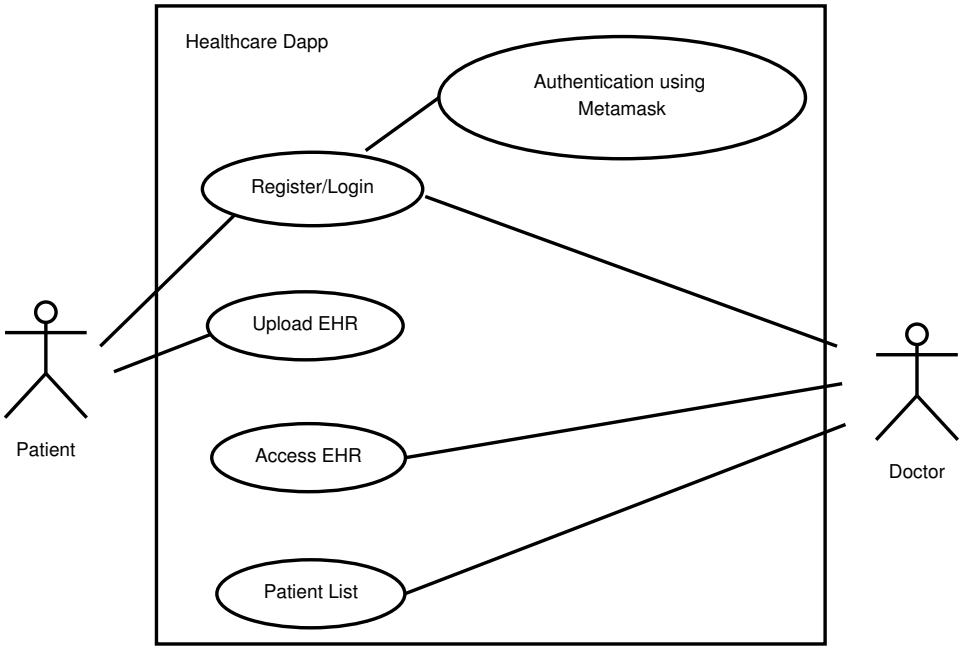

Figure 3.2: Use case diagram for the proposed system

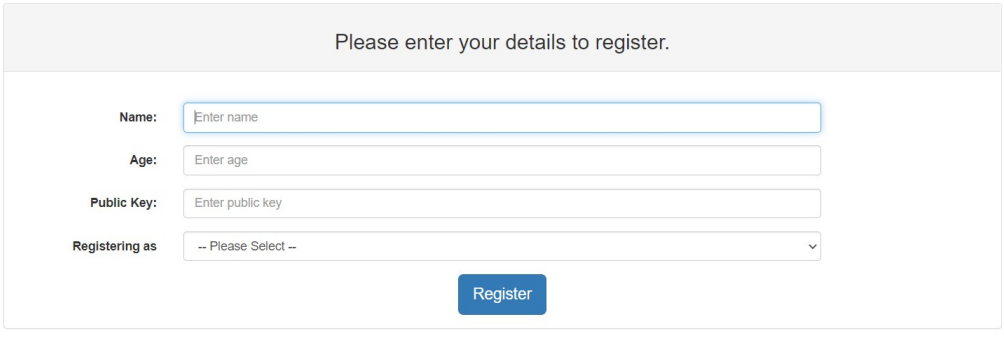

Figure 3.3: Attributes needed for the registration process

For the current work the number of doctors and patients are taken as 2. Figure 3.4 shows the generated doctor and patient account with their corresponding public key. For any kind of transaction in between doctor and patient this generated public key is used.

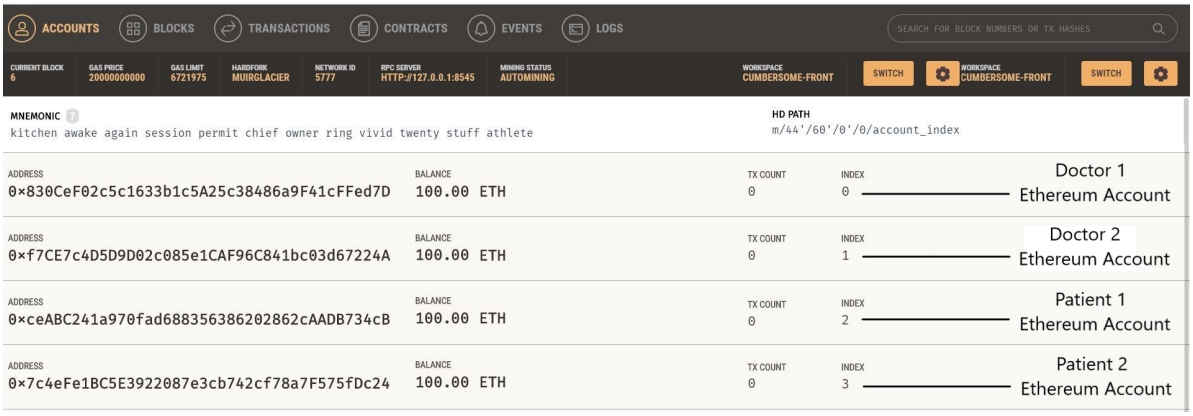

Figure 3.4: Ethereum account for doctor and patient with generated private key

3.6.1.2 Accessing EHR

After signing in the UI the patient has to choose the corresponding registered doctor for consultation. Once the doctor is making a diagnosis report then the report is considered as the EHR that needs to be stored in the blockchain network which can be accessed by any doctor chosen by that corresponding patient. Figure 3.5 and 3.5 shows the patient and doctor portal in the developed dApp.

3.6.1.3 Off-chain EHR storage and Payment

Once the registered doctor diagnoses the patient then the diagnosis report will be stored in the blockchain network in the off-chain storage manner. The patient can log in to the application to see the health diagnosis report and also can share the report with various doctors for further treatment. This process helps in avoiding the report storage at the patient side. These stored reports can be accessed by the doctor and the particular patient. Figure 3.7 shows the off-chain EHR storage of the users.

Upon successful diagnosis of the patient it has to make the payment to that corresponding doctor. For that, each user is attached to metamask through their private key. Metamask helps the patient to transfer the fee in terms of ETH to the doctor by using the public key of the doctor. Once the money is transferred the remaining balance is updated in the blockchain account balance of the user. For sending the money the user needs to use an amount of gas which is provided by the EVM. For the current dApp development, the *GanacheEVM* is used. Figure 3.8 shows metamask configuration for different users of

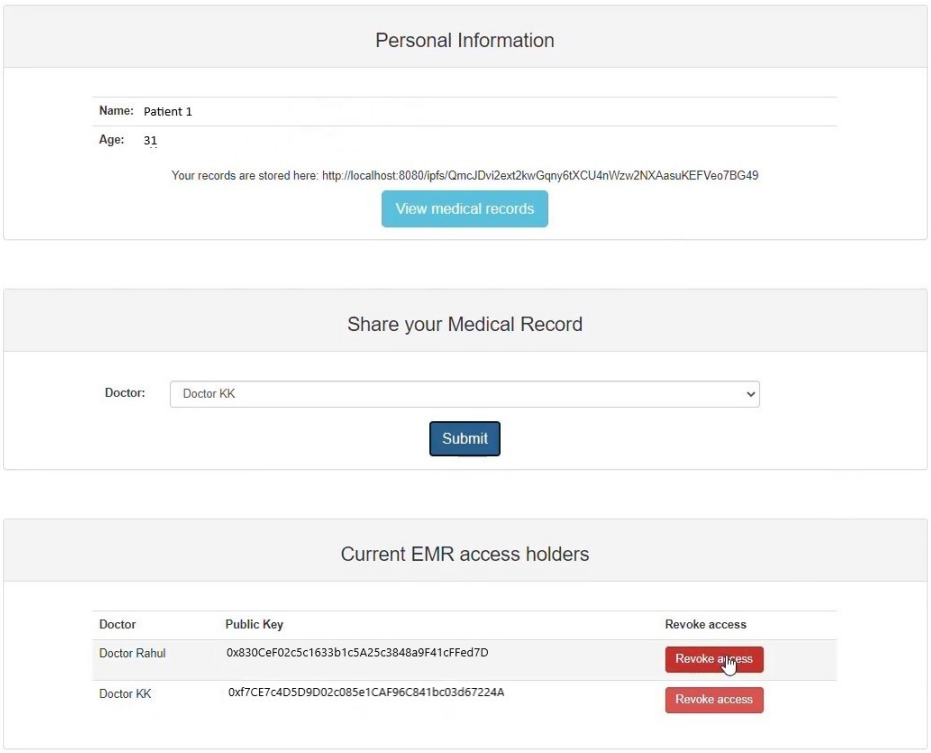

Figure 3.5: Patient dashboard for choosing the registered doctor

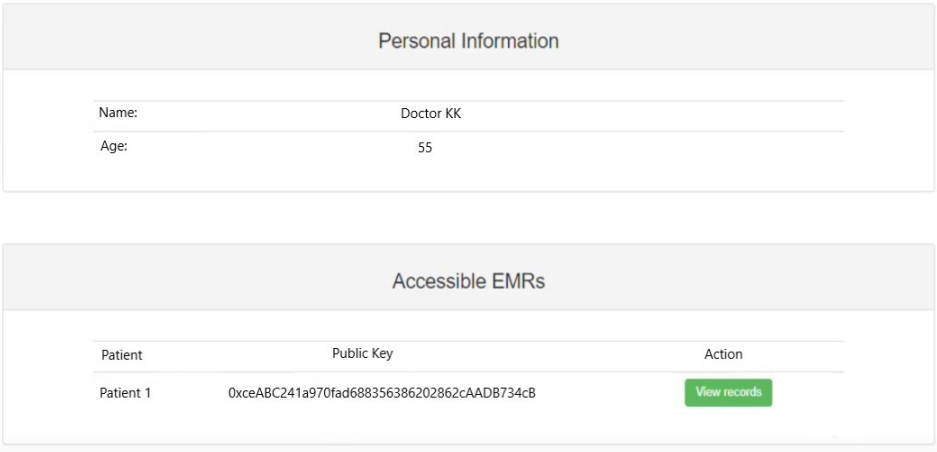

Figure 3.6: Doctor dashboard for choosing the registered patient

the current dApp. Figure 3.9 shows the updated ETH value in *Ganache* after successful transaction between the Doctor and Patient.

3.6.2 Crtical Analysis

The main benefit of the proposed dApp is the EHR sharing among the patient and doctor. The patient is able to share the previously diagnosed report with the desired doctor. The proposed model is compared with various existing models based on the adopted blockchain platform, key parameters, and payment options. Table 3 represents the comparative study.

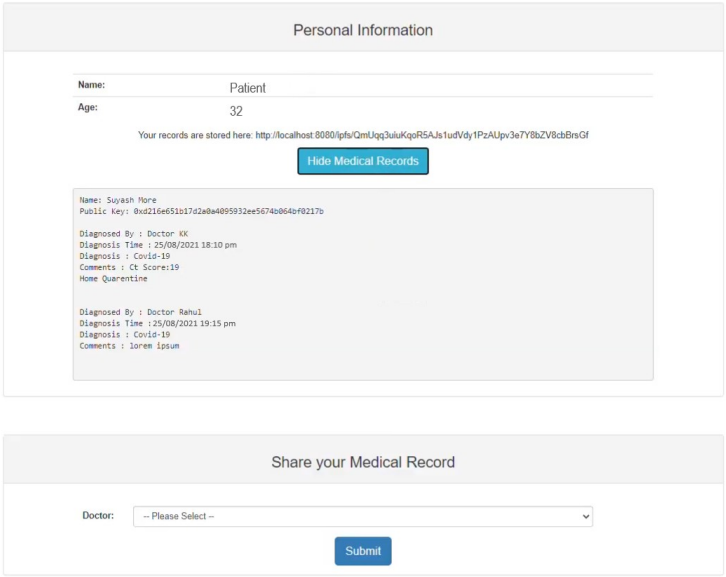

Figure 3.7: EHR off-chain storage of the registered patient

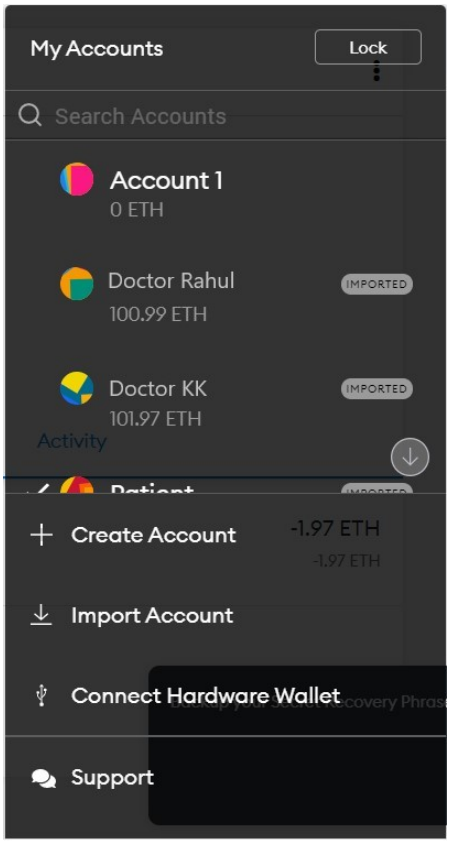

Figure 3.8: Metamask accounts for doctor and patient

Medrec [124] provides an Ethereum-based platform for storing and sharing patient medical data records. The limitation present in this work is that this model does not provide any mechanism to pay the consultation fee by using the Ethereum wallet. In

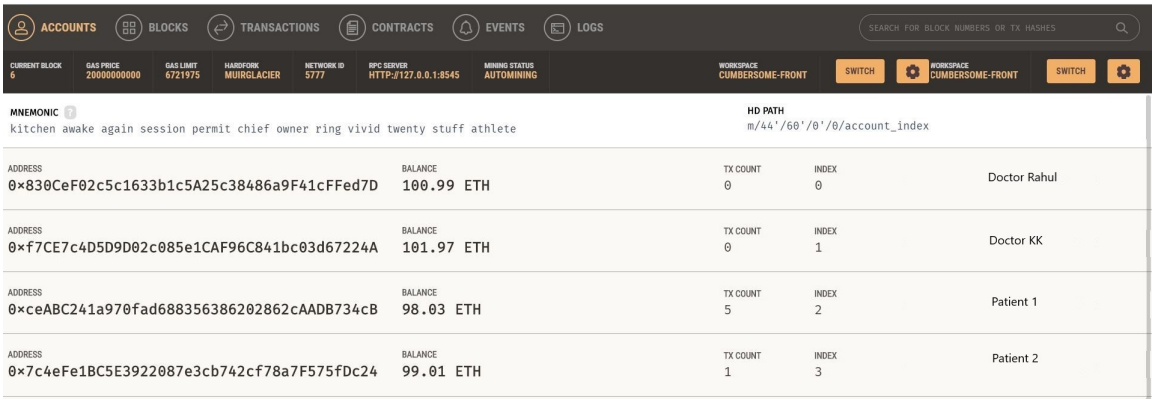

Figure 3.9: Ganache updated account balance in ETH after successful transaction

Table 3.3: Limitation of blockchain in the healthcare system

Model	Blockchain Platform	Use Case	Data Storage	Data Sharing	Payment Portal
Medrec [124]	Ethereum	EMR	Yes	Yes	NA
MIstore [125]	Ethereum	Security	No	No	NA
Remix [126]	Hyperledger	EMR	Yes	Yes	NA
Ancile [52]	Ripple	Insurance	Yes	Yes	Metamask
Proposed Work	Ethereum	EMR	Yes	Yes	Metamask

MIstore [125] a blockchain-based dApp has been proposed for data security. This model does provide any means to store and share the data. Remix [126] is a Hyperledge-based application system for storing and sharing the EMR but it does have any payment portal.

Ancile [52] is a Ripple-based application system for insurance claims. This application only allows the patient to store the EMR and share the same with the vendor providing the insurance. The Metamask wallet is used for insurance payments. The proposed work aims to provide an Ethereum-based dApp to store and share the data between the patient and doctor. The patient is free to make an appointment with any registered doctor and is also able to share the previously diagnosed report with the desired doctor. This application also allows the users to pay the consultation fee through the blockchain-enabled wallet meta mask where the user has to create an account by using the obtained private key.

3.6.3 Cost Estimation

For deploying the dApp in the real world the implementation cost needs to be defined. The main goal is to create a solution that can provide a viable healthcare system by taking advantage of blockchain's capabilities. Network exploitation and other computational concerns are prevented by taking some fee for any kind of transaction executed in the platform and the fee is set as gas and ETH. On the Ethereum blockchain technology, gas refers to the payment or price value necessary for a successful transaction or contract

execution.

For all kinds of computation done in EVM, the user needs to pay the fee. For every transaction initiated by different users, the gas limit needs to be set within which the user has to complete the transaction and the user has to return any unused gas to the network for which the user will be rewarded. If the users do not have sufficient balance in their account then it can initiate any further transaction. In EVM, ethers are used to buy gas, and users that are executing transactions can establish a gas limit for their account for that transaction. However, it is up to the miner to decide whether or not to allow the transaction. If a sender sets a higher gas price, it will cost them a lot of money to pay for the gas, while miners will gain a lot of money. The computation is then carried out by a miner to add this transaction to a block. A miner can then broadcast the new block into the network after all transactions have been completed successfully.

3.7 Conclusion

Blockchain system leverages the cryptography mechanism, P2P connection, consensus models, and smart contracts to build a decentralized communication and application. In this chapter, blockchain evolution has been focused on in terms of its application and feature. It is believed that dApps based on the blockchain can bring a new era to the application domain. In this current research work, the decentralized application for HCS is developed using blockchain technology for storing, sharing, and diagnosis purposes. The off-chain storage of this developed dApp deals with blockchain-based storage constraints. The current Healthcare EHR dApp will let the patient freely and securely share medical records with the doctors while maintaining access control and security.

The developed dApp only focuses on providing a simple blockchain application for information sharing between doctors and patients. However, the developed dApp does not focus on the security aspect of the blockchain network. Hence it may suffer from various blockchain network attacks.

Chapter 4

Smart Contract assisted Blockchain-based Public Key Infrastructure

The developed dApp in Chapter 3 only focuses on providing a simple blockchain application for information sharing between doctors and patients. However, the developed dApp does not focus on the security aspect of the blockchain network. Hence it may suffer from various blockchain network attacks. For the current work, a BC-PKI is developed to deal with different security related issue of the blockchain-based application system.

Public Key Infrastructure (PKI) is a reliable solution for Internet communication. The conventional PKI system is centralized, which exposes the infrastructure to many security issues. The digital certificate generation and validation processes in PKI suffer from high latency and inadequate authentication processes. Moreover, it needs enormous time and effort to mitigate the malfeasance of the Certificate Authority (CA). The complexity of employing the traditional key and certificate management increases by enforcing the centralized CA, which can compromise transaction security. [1]To overcome the aforementioned issues of PKI, three different solutions have been reported in the literature: Log-based PKI (LBPKI), Web of Trust (WoT), and blockchain-based PKI. The blockchain-based PKI achieves more attention as it is the combination of LBPKI and WoT, which serves distributed trust, log of transactions, and constant-sized data to verify the identity of users. Motivated by these facts, this chapter reports a blockchain-based PKI system that has a lighter smart contract and less storage capacity and is also suitable for lightweight applications. The lighter smart contract in our infrastructure uses a *threshold value*, which validates the limit of one participating node for becoming the CA of any transaction inside the network. This approach can prevent distributed de-

[1]The outcomes of this chapter was published in "ICADCML-2021“[C1], and “Transactions on Emerging Telecommunications Technologies" [J2]

nial of service (DDoS) attacks. This smart contract also checks the signer node address. The proposed smart contract can prevent seven cyber attacks, such as Denial of Service (DoS), Man in the Middle Attack (MITM), Distributed Denial of Service (DDoS), 51%, Injection attacks, Routing Attack, and Eclipse attack. The Delegated Proof of Stake (DPoS) consensus algorithm used in this model reduces the number of validators for each transaction which makes it suitable for lightweight applications. The timing complexity of key/certificate validation and signature/certificate revocation processes do not depend on the number of transactions.

4.1 Introduction

PKI is the primary building block of client-server communication over the internet. PKI defines a set of rules and protocols for the crypto algorithms: encryption, decryption, digital signature, and digital certificate verification process, which are used in secure communication. For server identity authentication, traditional PKI uses a digital certificate which is issued by a trusted third party named as Certificate Authority (CA). This certificate is a data package to identify the identity of the server. The digital certificate is associated with the public key, and it is protected by asymmetric key cryptography. The CA has three primary responsibilities (i) issuing, (ii) revoking (iii) distributing digital certificates. Therefore, it is the most crucial component of PKI. The digital certificate standard, ITU-T X.509[128] coheres to the public key with the DNS record.

The X.509 standard certificate provides a verification method for the private and public keys used for the communication. CA is the only component in PKI to validate a transaction. Traditional PKI system adopts a trusted third party for issuing the digital certificate for every transaction or communication over the internet. There are various third-party CAs reported in the literature, such as Comodo, IdenTrust, DigiCert, Certum, Entrust, etc[129]. The degree of the successful transaction between the client and server depends upon the correctness of the certificate issued by CA. The communications in the aforementioned PKIs rely on the third-party centralized CAs. If the CAs used in Comodo, IdenTrust, DigiCert, Certum, Entrust, etc., become malicious, then the entire communication will be compromised, and it leads to single point failure [130].

Comodo is the first CA which have suffered from cyberattacks. In 2011 it had issued nine fraud digital certificates to various domains. In the same year DigiNotar has issued around 600 fraud certificates to various organizations [131]. In the same year, the Dutch KPN CA has restricted itself from providing the digital certificate after being suffered from a DDoS attack [**?**]. All of the above scenarios clearly explain the drawback of the conventional PKI framework. The conventional CA is time consuming as numerous amount of vendors can choose a single CA, hence the CA will definitely require more time

to issue the certificates. CA has to validate every user for secure communication which makes the conventional PKI more expensive.

Despite single point failure [132], the conventional PKI system has several other drawbacks. The conventional PKI does not have any feature to detect compromised CA. Moreover, the complexity of key generation and key validation processes reduces the performance of the conventional PKI. Considering these threats, servers which are not able to secure their own identities satisfactorily cannot ensure that their communications are not compromised by a deceitful certificate which may cause Man in the Middle attack (MITM) [133].

The malicious certificate issued by a compromised CA can cause severe damage to the transactions of conventional PKI. A malevolent CA like in DigiNotar loses all of its trustworthiness, and it creates a rogue certificate, which makes the entire network at risk [130]. Therefore, the aforementioned statements brief four major concerns of conventional PKI:

- The trust of existing PKI is centralized to Certificate Authority (CA) which can cause single point failure.
- The communications governed by PKI rely on the third-party centralized CAs. The literature has reported many incidents of malicious CAs.
- There are no ways to detect malicious CA.
- The complexity of key generation and key validation processes reduces the performance of the conventional PKI.

Pretty Good Privacy (PGP)[134] is one of the cryptographic solutions against the issues stated above. Unlike traditional CA, PGP gives the opportunity to the participating node to verify the digital certificates of other participating nodes by including their corresponding signature. This attribute creates a trust model where every participating node becomes the verifier for the other. As stated above, the issues of conventional PKI systems are properly addressed by 3 different approaches, such as $Web\ of\ Trust$, *Log based*, and *Blockchain based* [135].

Web of Trust (WoT) is the first approach which addresses the centralization issue of conventional PKI. WoT allows the network participants to choose their own trustworthy certificate provider for transactions. This feature decentralizes the whole infrastructure. The crucial drawback of the WoT is the overhead of the new joinee. The selection process of CA in WoT network is very complicated, which makes it inappropriate for conventional applications. At each successful transaction, the CA increases its trust counter value. Thereafter, for the next transaction, the node chooses a validator which has the highest counter value. The counter value of a new joinee in WoT network is zero. Therefore, the

new joinee will never be selected as a validator of any transaction. This issue makes WoT unrealistic for PKI applications [136].

Public log used in Log Based PKI is one of the solutions which can monitor activities of the CA. The log server will be visible to the entire network. Any illegitimate digital certificate can be identified by this network, and the corresponding CA will be suspended due to its malicious activity [136]. The public log server used in Log Based PKI is always prone to single point failure issue, which is the main disadvantage of this infrastructure [137]. The literature also provides many blockchain based PKIs, which are discussed in Sec. 4.2 and Sec. 4.3.

4.2 Related Work

In the current section, several PKI solutions are discussed. The discussion includes a PKIs without blockchain technology in Sec. 4.2.1 and blockchain-based PKI solutions in Sec. 4.2.2.

4.2.1 PKI without Blockchain

This section discusses about existing PKIs frameworks which have not used blockchain.This type of PKI is further categorized into two groups: log based PKI (LBPKI) (4.2.1.1) and WoT based PKI (4.2.1.2).

4.2.1.1 LBPKI:

Certificate Transparency (CT) in articles [**?**] maintains a public log of all issued certificates which strives to alleviate the problem of incorrectly issued certificates. The public logs are auditable. Therefore, it is easier for any nodes to check different activities like new certificates generation and certificate deletion. The public logs do not eliminate the risk of certificate misuse. It does not guarantee that the user is able to notice certificate misuse when it occurs.

Proposed Accountable Key Infrastructure (AKI) [138] is used to defend domains and clients from flaws induced by single points of failure. The check and balance method in AKI distributes the trust properly among multiple parties including CAs and domains. Even if the domain key is lost or breached, the AKI executes routine certification processes effectively and gracefully. It was presented as a solution for a public-key validation infrastructure. It selects a set of trusted nodes for validating the entire transactions in the network which decreases the dependency on any one node.

Attack Resilient Public-Key Infrastructure (ARPKI)[139] makes all of the certificated-related computations such as (i)certificate issue, (ii)update, (iii)revocation, and (iv)validation

processes transparent. ARPKI starts working with 2 different parts. The first part contains two different CAs and the second part contains one Integrated Log Server (ILS) for performing any operations. It ensures that the security will be preserved, even if the $n-1$ nodes are compromised out of all n number nodes.

Policert[**?**] is a broad log-based and domain-oriented architecture which uses a more secure authentication process for securing the domain's public keys and an extensive certificate management method for validating the transaction.

4.2.1.2 WoT based PKI :

LOCALPKI [140] was developed for the Internet of Things applications. In this PKI a local authority binds the public key with the user identity and the certificate is issued by a third-party node or local authority. A third-party entity is used in LOCALPKI to record this binding information and to provide registration updates.

The Notary-based PKI (NBPKI) [141] approach creates a group of trustworthy individuals known as Notarial Authorities (NA). The NA confirms the reliability of a certificate for validating a certain signature at a specified time. The end users depend on NA's public keys and self-signed certificates for producing and validating signatures. The working principle of NBPKI relies on three different components (i)end-user, (ii)Registration Authority (RA), and the Notarial Authority (NA). The end-user needs to register with RA for signing their transactions. The RA verifies the end-user identity and informs the associated NA. The NA decides the status of the trustworthiness of the end-user based on the information provided by the RA.

4.2.2 Bloockchain based PKI

This paper primarily addresses 8 attributes to compare different PKI system such as feature, type of blockchain network, blockchain platform, certificate, trust model, off-chain storage, on-chain and time complexity. Table 4.1 shows the detailed study of different blockchain based PKI systems.

- **Key Feature:** It shows the basic characteristic such as smart contract, CA, public ledge, etc. The blockchain based PKI is developed based on these key features.
- **Blockchain type:** The adopted blockchain network can be either of permissioned or permissionless blockchain. In a permissioned network, the new node can only join when it gets permission from every participating node present in the network whereas, in the permissionless network, new nodes do not require permissions from other nodes exist in the network. Instead of that, it takes permission either from one trusted node or from anyone randomly chosen node.

- **Blockchain Platform:** It shows the platform on which the PKI is implemented. The platform can be on the shelf platform such as Ethereum or a self-developed custom platform. The shelf platforms are publicly available and it needs to be downloaded from a trusted source and configured as per the requirement.
- **Certificate:** It shows the type of certificate used during the PKI development. It can be a X.509 standard or a custom one.
- **Trust Model:** It represents the mechanism for selecting the CA for validating a transaction. One node can choose a trustworthy node or a random node who solves the NONCE first.
- **Consensus Model:** It shows the adopted consensus model during the PKI development.
- **Storage:** The blockchain data can be stored in two forms such as the entire copy of the data will be stored, or the hash function of the block will be stored. There are two categories present for blockchain data storage named as on-chain storage and off-chain storage. On-chain storage allows the node to store the data directly on the blockchain network. Whereas the off-chain storage allows storing the data in a public ledger that is accessible by all other nodes or in a private storage from which that particular node can access it.
- **Time Complexity:** This shows the algorithmic computational complexity in terms of time. It has been taken in big O format as for every PKI all of the defined methods needed to be executed for a successful transaction. So the worst time complexity has been considered for different available blockchain PKI.

4.3 Problem Statement and Motivation

The trust of the traditional PKI systems completely depends on third-party CAs. The CA checks the bindings between public keys and entities and then provides digital certificates to those entities. A digital certificate assures that a CA confirms the binding process [**?**]. There are a very limited number of CAs that are trusted by modern browser and OS manufacturers. Therefore, this CA-based PKI architecture is considered a centralized infrastructure. The present CA-based PKI architecture, such as CT [**?**], AKI [138], and ARPKI [139] have adopted many methods to reduce the dependence on the confidence of CA. The primary concern in adopting those PKIs is to avoid the centralization issue of the infrastructure.

Table 4.1: Comparative study of existing blockchain based PKI systems based on the defined features

PKI	Key Feature	Blockchain Type	Block-chain Plat-form	Certificate	Trust Model	Consensus Model	Off-chain Storage	On-chain Stor-age	Time Complex-ity
PA-PKI [142]	Identification and verifica-tion of CA	Permissioned	Ethereum	Custom	WoT	PBFT	Private	Hash	$O(n)$
Block-CAM [143]	CA in cross domain veri-fication	Consortium and Permis-sioned	Ethereum	X.509 v3	Hierarchical	NA	Public Data	Hash + Data	$O(n^3)$
BC-TRUST [144]	Authentication	Permission Less	Ethereum	Custom	WoT	NA	Public Data	Hash + Data	–
BLOCK-PGP [145]	Access Con-trol of Cer-tificate Revo-cation	Permissioned	Ethereum	X.509	WoT	PoW	Public Data	NA	–
PB-PKI [146]	Public Ledger	Permission Less	Custom	Custom	WoT	NA	Private Data	Hash	O(n)
TTA-SC [147]	Automating the process of identifying the miscon-figured CA	Permission Less	Ethereum	X.509	Hierarchical	NA	Public Data	Hash + Data	$O(n)$
CERT-CHAIN [148]	CA Trust-worthy by using Dual Counting Bloom Filter (DCBF)	Permission Less	Custom	X.509	Hierar-chical	Dependa-bility rank based	Public Data	Hash	$O(n^2log(n))$
CERT-LEDGER [149]	Certificate Trans-parency	Permission Less	Ethereum	X.509	Hierarchical	PBFT	Public Data	Hash	$O(log(n))$
DB-PKI [150]	CA	Permission Less	Custom	Custom	WoT	PBFT	Public Data	Hash	$O(n^2)$
IKP [151]	CA trustwor-thy	Permission Less	Ethereum	X.509	Hierarchical	NA	Public	Hash +Data	$O(nlog(n))$
FLY-CLIENT [152]	Transaction Verification for light client	Permissioned	Ethereum	Custom	Hierarchical	PoS	Public	NA	$O(logn)$
BLOCK-QUICK [153]	Transaction Verification for light client	Permission Less	Ethereum	Custom	WoT	PoPoW	Public	NA	$O(n)$

Blockchain Based PKIs such as PA-PKI [142], Block CAM [143], PB-PKI [146] etc. provide an emerging alternative for conventional PKI systems which adopts different features of Log based and WoT approaches. Blockchain-based PKI provides an environment for decentralized authentication and validation of transactions in the network [16]. The adoption of different CAs for different transactions in Blockchain-based decentralized PKIs eliminates many issues caused by legacy PKIs. The use of different CAs for different transactions increases the fault tolerance capacity of the network and one malicious CA can not sabotage the entire chain.

The distributed Log in blockchain-based PKI provides a certificate transparency feature that is similar to the certificate transparency (CT) characteristic provided by Google

which helps to improve the security of PKIs. The CT allows logging and observing the scope of digital certificates. Examples of blockchain-based PKI systems are Namecoin and Emercoin [154]. Namecoin and Emercoin need enormous storage for the entire blockchain information for validation purposes and they also need to store the entire blockchain copy at the user end. These storage issues have made these blockchain-based PKI impractical for real-life applications. The smart contract-based PKI simply dissociates the storage from the validation process where one node does need not to store the entire blockchain copy for validating a transaction [155]. The major lacunas of existing blockchain-based PKIs are :

- All the participants in existing blockchain-based PKI do not get a fair chance to become CA.
- This complexity of the consensus algorithm in blockchain-based PKI makes it inefficient especially for lightweight applications.
- Most of the blockchain-based PKIs have concentrated on Denial of Service (DoS) and Man in Middle Attacks (MITM). They have not addressed Distributed Denial of Service (DDoS), 51% attacks, Injection attacks, Routing Attacks, and Eclipse attacks.

4.4 Objective

The objective of this work is categorized into two subsections as follows:

- **Approach 1**
  - To develop a simple blockchain-based PKI using the smart contract.
  - To evaluate the proposed model by calculating the lapse time for generating and validating the key pair, and gas utilization needed for a transaction.
- **Approach 2** It is the extended version of **Approach 1** with the following additional functionality to deal with various blockchain network based attacks.
  - To modify the **Approach 1** to prevent DoS, DDoS, MITM, 51%, Injection, Routing, and Eclipse attacks. The proposed smart contract checks the validity of the signer node address and it also imposes a threshold value for becoming CA which gives a fair chance to all the participants to become CA.
  - This work adopted Delegated Proof of Stake (DPoS) consensus algorithm which reduces the number of validators of each transaction. Therefore it reduces the timing complexity which makes it suitable for lightweight applications.

– The proposed PKI system is evaluated based on the two matrices. (i)lapse time of key generation and key validation process and (ii) gas cost of the transaction. The result shows the time complexity of the proposed blockchain-based PKI system is efficient compared to existing literature.

4.5 Methodology

The proposed smart contract based PKI system is implemented in the open-source Ethereum platform known as the Go Ethereum or GETH. Ethereum is an open-source platform where the smart contract is the key functionality. It provides a virtual environment where multiple live nodes are deployed to create a blockchain network. Smart contracts are written in Turing complete language known as Solidity which is executed in the Ethereum virtual machine [156]. The smart contract code is publicly available to all participating nodes present in the blockchain network. In the current research work, multiple nodes are deployed with some initial cost and gas using GETH. For every transaction, the node needs to share some gas (G) and each gas has some price (P). So, the total cost (C) in terms of ether (ETH) can be expressed as equation4.1. Section 5.6.1 and 5.6.2 show the methodology description for Approach 1 and Approach 2 respectively.

$$C = G \times P \tag{4.1}$$

4.5.1 Methodology for Approach 1

The PKI framework consists of two primary components as Smart contract and the participant. The Smart contract is nothing but the set of protocols needed to meet a common agreement while storing the transaction records where as the participant is the node that will take part in the communication. For deploying the smart contract the Ganache is being used where 10 participant node has been created to form a P2P network. Each node has been assigned a unique address with some initial currency as 100 ETH.

4.5.1.1 Model Description

Participant, *Signature*, and *Revoke* are three different modules used to develop the smart contract based PKI system The working of these modules are as follows:

- **Participant** This class or method will allow the creation of new participants in the peer-to-peer network. Once the participant node has been created then the following attributes will be assigned to it. Whenever it is invoked in the contract it is stored in an array from which the node can be accessed by calling the ID. The participant ID is the array index.

- ID: Id is a unique number that helps in identifying the particular participant node during the transaction.
- Ethereum Address (ETH Add): It will specify the address of the node inside the network.
- Key Pair: It will generate the key pair of all participating node in the network.

- **Signature** This will allow the participant node to sign the public key certificate for other nodes. For signing the following attributes have been defined.
  - Sign id: The Ethereum address of the signing node is stored here.
  - Key: It provides the public key which is required to sign the data.

- **Revoke Signature** This method will help the participant node to revoke its own signature. It has been defined with the following attribute.
  - Revocation Id: It is an identifier number that will indicate the number of times the signature is being revoked.
  - Counter: It is the counter for counting the number of time the node becomes the CA.

4.5.1.2 Working Principle

In the proposed framework there are three primary modules such as Participant, Signature, and Revoke. Before beginning any transaction, the node status will be checked and the transaction is allowed id the node is already present in the network. However, if the node is found new then the *Particpant* module is invoked. The *Participant* module sets all the required parameter for a node such as ID, $ETHAdd$, and key pair.

Once the node parameters are set successfully then the signer node or CA is fixed using WoT model. For validating the transaction the *Signature* module is invoked. After each successful the counter is increased by 1 to keep track the number of times that particular node is becming the CA. The detailed workflow of the proposed work is depicted in 4.1.The pseudocode for the proposed system has been specified in Algorithm 1.

4.5.2 Methodology for Approach 2

The proposed smart contract based PKI system is implemented in the open-source Ethereum platform known as the Go Ethereum or GETH. The main building blocks of the proposed PKI system are smart contract and Ethereum. Ethereum is used as the platform where the smart contract is the core part of the work.

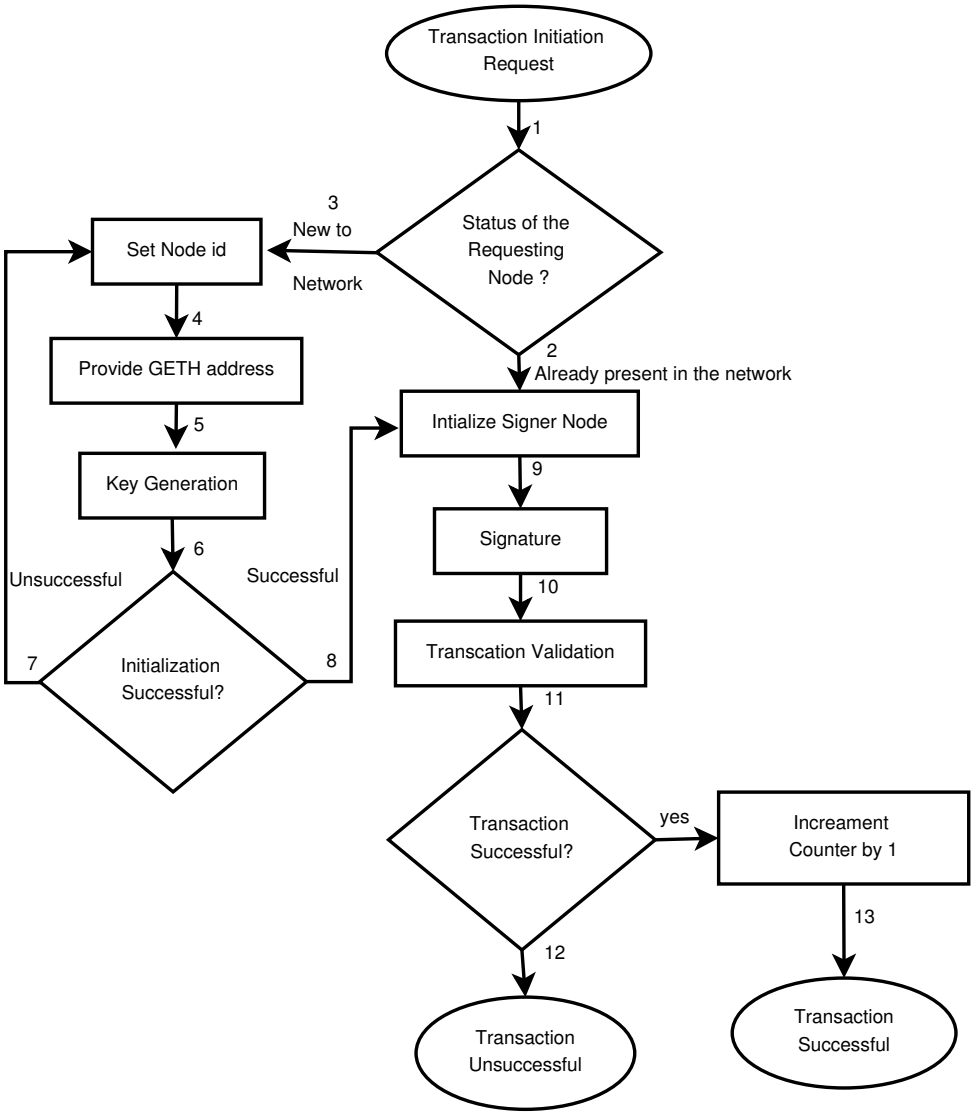

Figure 4.1: Working of proposed smart contract based PKI

4.5.2.1 Model Description

The proposed PKI system contains three basic modules such as Participant, Smart Contract, Signature, and Revocation. The participant module contains the method to add the attributes of a participating node when it is new to the network. The signature module enables the nodes to sign and validate the keypair. The revocation module allows the node to revoke its own signature so that the corresponding node can resign another transaction.

- New Participant: The input of this module is the status of the node. If the node is found as a new node of the network, then the 3 attributes: PID, $ETH\ address$ and $Keypair$ will be set to the status of the new node to participate in the transactions of the network. If a node already exists in the network, the participant module invokes the aforementioned attributes to participate in the transaction. The pseudo-code for

Algorithm 1 Algorithm for Smart Contract Invokation in PKI

```
  BEGIN TRANSACTION
Require: Node.stats
  PROC PARTICIPANT()
  get Node.stats()
  if (Node.stats==FALSE) then
      Set id
      Set Private key (pr_key) and Public key (pb_key)
      Set id.getGethAddress()
  else
      Node is present in the Ethereum private network
  end if
  PROC SIGNATURE ()
  Signer.id
  get Pb_key
  PROC REVOKE()
  if (Trnsaction is Successful) then
      Counter=Counter+1
  else
      Abort
  end if
```

this module is presented in algorithm 2. The attributes of the participant module are stated below:

- ***PID:*** It is a unique random number that can be used to identify a particular node in the network.
- ***ETH address:*** It is an address provided by the Ethereum blockchain environment which is required during transactions.
- ***Keypair:*** The private and public key pairs will be generated and assigned to a particular node.

As the current research considers a lighter smart contact, only the PID of that corresponding node is stored after deployment.

- **Smart Contract:** The inputs to this module are the PID, RID and $ETHadress$. The PID and the $ETHaddress$ of the chosen signer node are compared with the stored PID and $ETHaddress$. If both of the addresses are matched then the RID of the signer node will be compared with the defined threshold for that node. The transaction will be allowed only after the successful execution of the above said conditions. The detail pseudocode is reflected in algorithm 3.

Algorithm 2 New Participant

BEGIN TRANSACTION
REQUIRE: **Set of Nodes N=[**N_1,N_1,N_1,.....,N_n**]**
PROC ***PARTICIPANT***()
get N_i.status
if (N_i.status==FALSE) **then**
    set PID
    set (PR_{key}, PB_{key})
    set PID.getETHAddress()
    set PID.Limit
else
    Node N_i is present in the Ethereum private network
end if
run CONTRACT()

Algorithm 3 Smart Contract

TRANSACTION PROCESSED
REQUIRE: N_i**.RID,**N_i**.PID,**N_i**.ETHAddress**
get Signer.PID
get Signer.RID
get Signer.ETHAddress
if (Signer.PID==N_i.PID *and* (Signer.ETHAddress==N_i.ETHAddress) **then**
    if (N_i.RID $\leq$ N_i.limit) **then**
        PROC SIGNATURE
    else
        Maximum Trial is over for the elected signernode. Please select another node
    end if
end if

- **Signature Validation:**This module allows the nodes to sign the transactions of the other nodes. When the node is elected as the signer node, this method will be called with two attributes such as the PID and Expiry. The steps are shown in algorithm 4.
  - ***PID:*** It is the unique number assigned by the Participant method which provides the unique identity.
  - ***Expiry:*** After the validation process the node needs to increase the predefined counter by one to ensure that all of the participant nodes present in the network will get an equal chance to become the transaction lead. This counter value is the maximum number for which one node can be elected as the transaction lead. In the current research work, it is defined in the smart contract to avoid the DDoS attack.

- **Revocation:** It is called by the leader node after every transaction. It contains the

Algorithm 4 Signature Validation

TRANSACTION PROCESSED
REQUIRE: N_i**.PID,** N_i**.ETHAdress**
PROC SIGNATURE ()
get Signer.PID
get Singer.ETHAdress
validate(TRANSCATION)
PROC REVOKE()

counter described in the signature module. The node increases the counter by one after every successful transaction. If the counter exceeds the maximum limit defined in the light version of the smart contract, the election process is rejected and the process is reinitiated. Revoke ID or RID and Signer ID are two attributes present in this module. The pseudo-code for this module is represented in algorithm 5

- ***RID:***It is a counter which is increased by the leader node after the successful completion of the transaction.
- ***Signer ID:*** It is the id of the node which is going to validate the transaction.

Algorithm 5 Signature Revocation

TRANSACTION PROCESSED
REQUIRE: N_i**.RID**
if (TRANSACTION==TRUE) **then**
    RID ++
else
    Transaction is rejected
end if

4.5.2.2 Block structure

Each block has 2 components: block header and list of transactions. Block header has 3 fields: (i) Block root hash, (ii) Hash of the previous transaction, and (iii) Markel Patricia Tree (MPT). Figure 4.2 represents the structure of block where n is the number of transactions. Here T_1 to T_{n-1} are previous validated transactions and T_n denotes the current transaction. H_i denotes the hash value of T_i where i varies from 1 to n. The number of transactions stored in a single block may vary with different blockchain platforms. The size of blocks on certain blockchains, such as Bitcoin, is limited. The 'genesis block', or the first block on the blockchain, is noteworthy. It has no hash that refers to a parent block, and it does not allow any mining process. Blocks are issued at fixed intervals. In current Ethereum blockchain new blocks can be released at every 15 seconds interval. The merkel tree has three type of nodes: (i) Leaf Nodes (H_1, H_2, H_3, ... H_n) (ii)Intermediate

Nodes ($H_1||H_2$,...$H_{n-1}||H_n$) and (iii) Root Nodes ($H_1||H_2||...||H_{n-1}||H_n$). These hashes are also used as the node's reference key. The leaf node (L_i), intermediate node (I_i), and root node (R) of the MPT are defined as in equations 4.2,4.3 and 4.4 respectively.

$$L_i = H_i = hash(x_i), \{i\epsilon 1, 2, 3, 4, ..., m\} \tag{4.2}$$

$$I_i = \{H_i \parallel H_{i+1}\} \tag{4.3}$$

$$R = \{H_i \parallel H_{i+1}... \parallel H_N, N = Depth of MPT\} \tag{4.4}$$

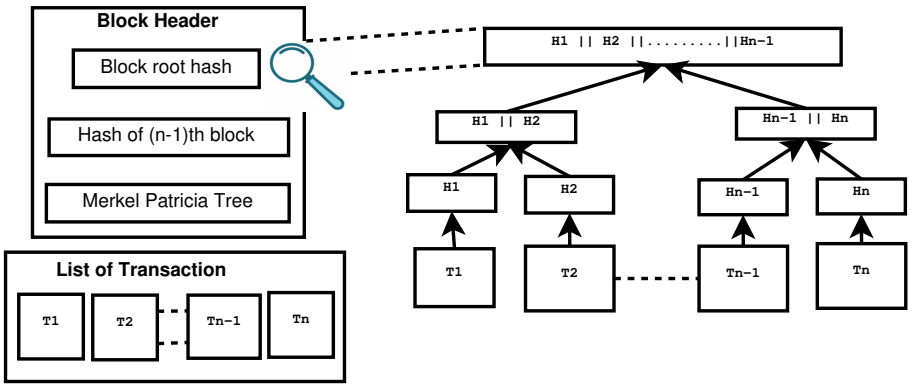

Figure 4.2: Working of proposed smart contract based PKI

4.5.2.3 Delegated Proof of Stake Consensus Mechanism

The Delegated Proof of Stake (DPoS) [**?**] consensus algorithm is a variance of the PoS mechanism which improves scalability and efficiency by lowering and limiting the number of validators on the network. It was designed to address the issue *scalability trilemma* [**?**]. In blockchain terminology, the more number of transactions per unit time refers to more scalability. As per the blockchain trilemma, more scalability may cause more challenges for security and decentralization features. In DPoS, token holders do not work on the validity of the blocks directly; instead, they choose delegates to validate transactions on their behalf. There are typically 21–100 designated delegates in a DPoS system. The chosen delegates are rotated regularly and the nodes order the delegates to present their blocks. When there are fewer delegates, it is easier to allocate one validator and time slot for each transaction. If the delegates consistently miss to validate transactions or blocks, it will cause erroneous transactions. As a result, the token holders vote them out and replace them with another delegate chosen by the token holders.

4.5.2.4 Working Principle

Once it receives the transaction request, the participant module starts its execution to check the status of the node. If the node is found as a new node, the required parameters

such as the PID, $ETH\ Address$, $keypair$, and a threshold value for RID will be specified for the node. This RID is incremented by one in each revocation call and once it reaches to the threshold the PID, $ETH\ Address$ and $keypair$ of the node will be reset. The PID and $ETH\ Address$ identify a particular node at any time uniquely.

After the successful execution of the participant module, the smart contract is invoked. Thereafter the PID of the selected signer node is compared with the stored PID. If both PIDs are matched further execution will be allowed otherwise the process will be aborted. Then the RID counter will be compared with its threshold limit. If the RID exceeds the given threshold, the transaction will be aborted immediately otherwise, the signature module will be invoked. The adoption of the smart contract in our methodology helps the network to deal with the DDoS and MITM attacks by verifying the node id and checking the limit respectively.

The signature module allows the selected signer node to validate the transaction by verifying the public key. The Signature module allows that particular node to validate the transaction which completes the smart contract verification phase.

After every successful transaction, the signature revocation module is invoked where the signer node increments its RID value by 1 and validates the transaction. Figure 4.3 represents the workflow of the proposed work.

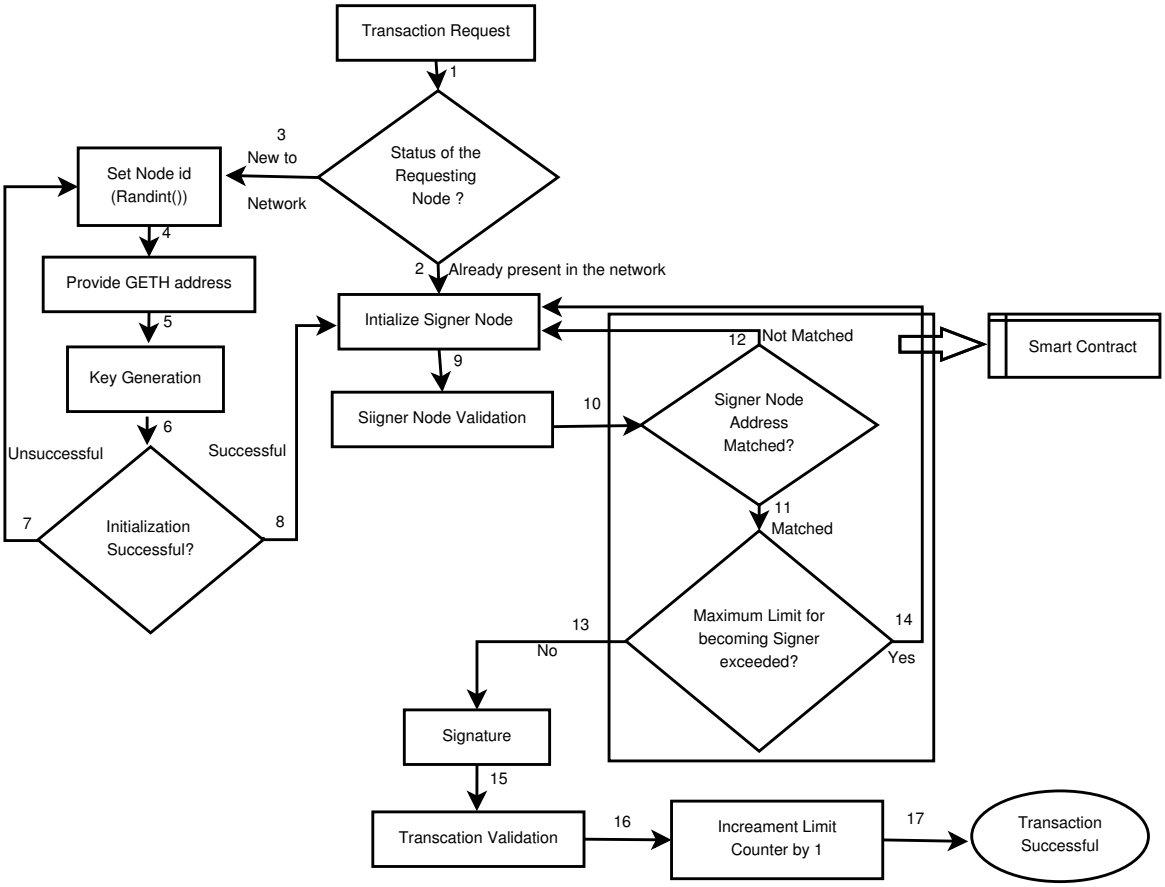

Figure 4.3: Workflow of proposed Blockchain based PKI

4.6 Implementation and Performance Evaluation

The proposed works (**Approach 1, Approach 2**) is implemented in the open-source Ethereum virtual machine $GETH$. To invoke the smart contract, the $Solidity$ $v0.4.24$ scripting language is used along with the $GANACHE$ $truffle$ suit. The $truffle$ suit deploys the developed smart contract in the blockchain environment. The experiment is carried out with a Windows 10 OS, 8 GB RAM, 1 TB HDD, and $Intel$ $i5$ processor with a 2.8GHz clock speed machine.

4.6.1 Performance Evaluation of Approach 1

For the the intially developed PKI (Approach 1) 5 different network has been created with several nodes as 5,6,...9. Each node has been supplied with 100 ETH as primary balance which can be used during the transaction. The number of transaction considered for the current PKI is 100. The GANACHE truffle suit is used to deploy and test the smart contract in a deterministic environment. GANACHE will help us to visualize the transaction details smart contract deployment details and as well as the created node accounts. Finally, to measure the performance of the proposed algorithm the Lapsed time for generating the keypair and validating those will be measured. The measured time has been shown in Table 4.2 and in Figure 4.4. It shows that increasing the number of nodes the latency will also increase. The latency will increase because the number of the trustworthy node will increase for every node, so during the transaction key revoking the request will be forwarded to every trustworthy node which will result in increasing the latency.

Table 4.2: Time Lapse for creating and validating tables

Number of nodes	Time Lapse for Key Generation (in min)	Time Lapse for key Validation (in min)
5	1.89	1.93
6	2.31	2.69
7	2.89	3.83
8	4.17	4.87
9	5.78	7.41

Figure 4.5 shows gas utilization vs the number of transaction graph where the average gas cost for each transaction is approximately 2.3×10^4. Table **??** shows the gas used for invoking different modules of the current PKI system.

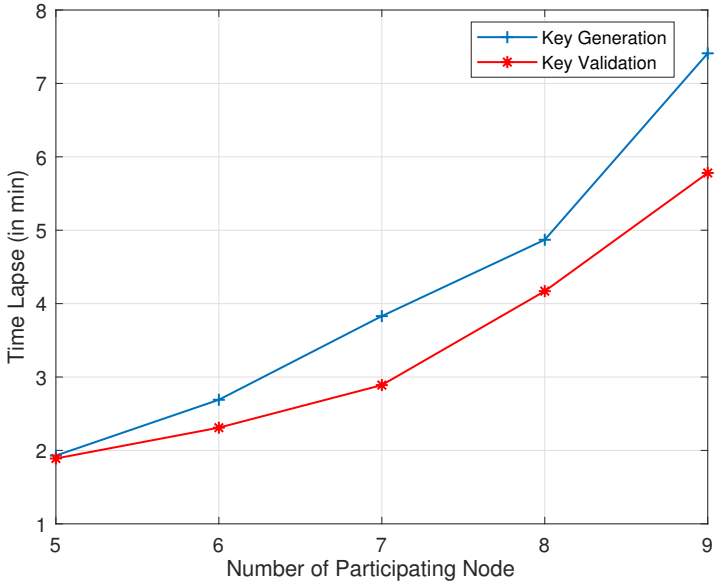

Figure 4.4: Latency plot with different number of blockchain nodes

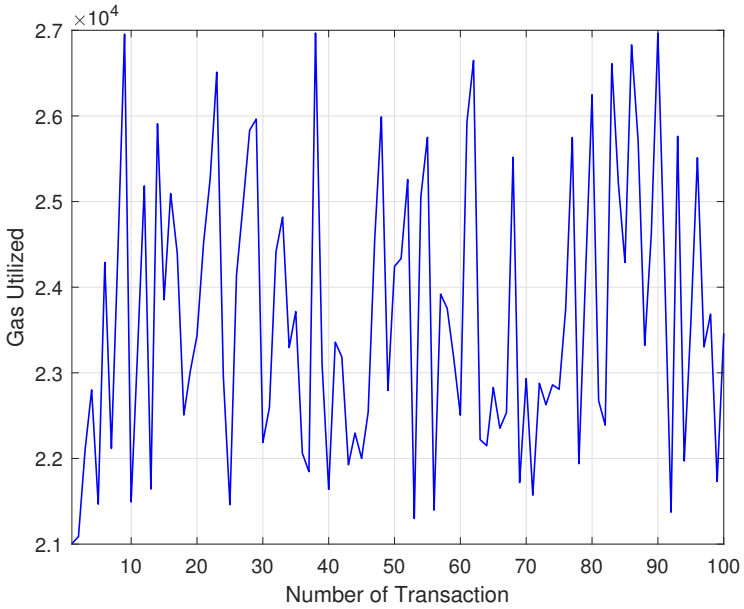

Figure 4.5: Gas utilization for different transaction

Table 4.3: Gas usage for invoking various modules

Module Name	For Initialization
Participant	23675
Signature	25913
Revoke Signature	21000

4.6.2 Performance Evaluation of Approach 2

The proposed work is implemented in the open-source Ethereum virtual machine $GETH$. To invoke the smart contract, the $Solidity$ $v0.4.24$ scripting language is used along with the $GANACHE$ $truffle$ suit. The $truffle$ suit deploys the developed smart contract in the blockchain environment. Initially, the Gas limit of the network is set as 4000000 and all created nodes have $100ETH$ in their account.The performance of the proposed PKI system is evaluated using the latency and gas utilization during the transaction. Figure 4.6 shows the node iitialization in the $GETH$ environment. The node configuration with the private key and $GETH$ address is shown in Figure 4.7.

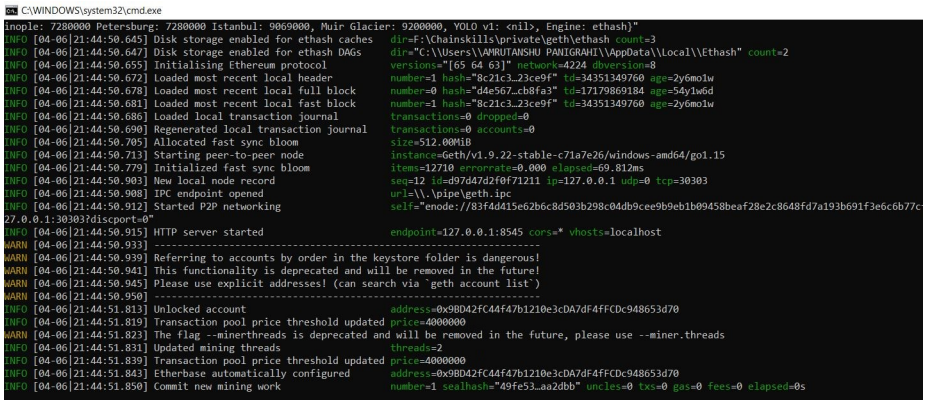

Figure 4.6: Node initialization in $GETH$ environment

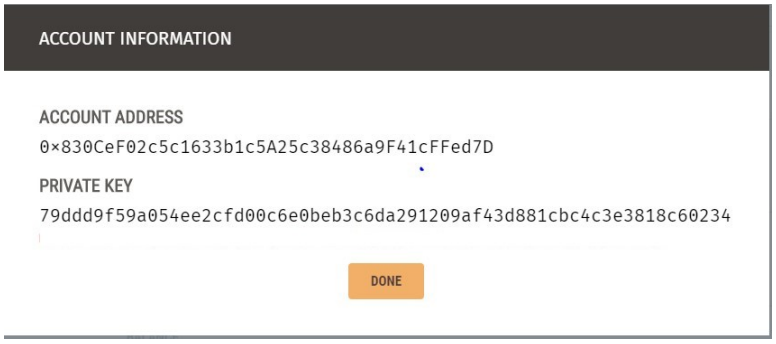

Figure 4.7: Node Configuration

Figure 4.8 shows the latency vs the number of nodes graph for key generation and key validation process. The proposed model is tested with 100 nodes where latencies of key generation and key validation process reach to 60 seconds and 80 seconds respectively which is suitable for realistic applications of PKI.

Table 4.4 shows the gas used by the different modules of the developed PKI system for doing one transaction. Figure 4.9 shows gas utilization vs the number of transaction graph where the average gas cost for each transaction is approximately 10×10^4.

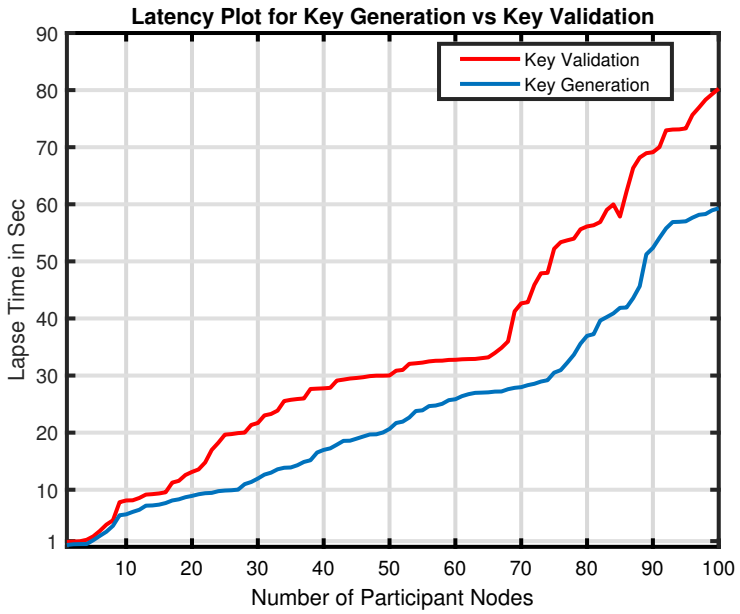

Figure 4.8: Latency vs Number of Nodes for Key generation and Key validation Processes

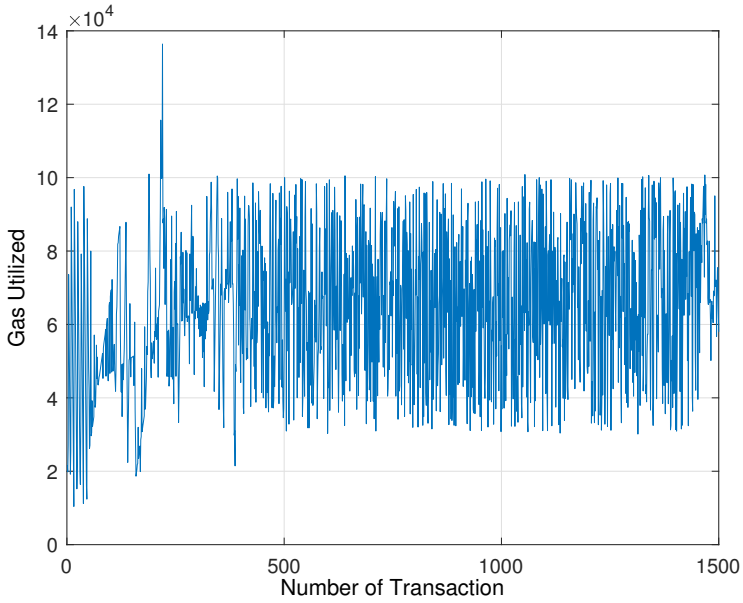

Figure 4.9: LGas utilization vs Number of Different Transactions in the Network

4.6.2.1 Time complexity Evaluation

There are four executable modules present in the developed blockchain based PKI system, namely *participant*, *signature*, *revoke*, and *smart contract*. Among these four modules, the time complexity of *participant* and *smart contract* module is $O(n)$, whereas the time complexity of *signature* and *revoke* modules are $O(1)$. Here n is the number of transactions committed to the procedure in the network. Multiple transaction requests

Table 4.4: Gas usage by various modules

Method Name	Gas Utilized	
	For Initialization	For Transaction
Participant	33781	17484
Signature	42856	13752
Revocation	19798	9689
Smart Contract	194837	32675

may be raised in the case of participant and smart contract module resulting in the worst time complexity of these two modules as $O(n)$. While there is no communication in the other two modules: *signature* and *revoke* and also, no acknowledgment messages are issued to the transaction initiator. The *signature* and *revoke* modules allow the chosen signer node (by smart contract module) to sign the transaction and make an increment of RID. So these two procedures do not generate any transaction messages, which results in constant time complexity of $O(1)$. Implementing the DPoS consensus mechanism results in a run time complexity of $O(logn)$. The time complexity of the whole system is $O(n + logn)$. The time complexity of the proposed model is compared with the different exiting models in Table 4.5.

Table 4.5: Module wise Time Complexity Comparison with different existing models

Blockchain based PKI	Key/ Certificate Generation	Key/ Certificate Validation	Signature/ Certificate Revocation
PA-PKI [142]	–	$O(n)$	$O(n)$
CERT- CHAIN [148]	$O(n^2)$	–	$O(log(n))$
CERT-LEDGER [149]	$O(log(n))$	–	–
DB-PKI [150]	$O(n^2)$	–	$O(n^2)$
FLY-CLIENT [152]	–	$O(logn)$	–
BLOCKQUICK [153]	–	$O(n)$	–
Proposed System	$O(n)$	$O(1)$	$O(1)$

4.6.2.2 Critical Analysis

This work addresses various limitations of the existing PKI solutions including PKI without blockchain stated in Sec. 4.2.1 and PKI with Blockchain stated in Sec. 4.2.2. The PKI provided in [**?**] only focuses on making the issued certificate visible to the network participants but does not have any circumstances to avoid the single point of failure (SPoF) limitation. In AKI [138] the ILS is responsible to store the certificate issued by CA and the ILS will be updated at a given time interval even CA becomes untrusted. This becomes the key limitation along with SPoF as it is using a centralized CA to issue the certificate. ARPKI [139] tries to solve the synchronization issue of AKI but it still depends upon a trusted CA to issue the certificate. The unavailability of the CA verification process makes it tough to adopt ARPKI as a preferred solution. The approach in PoliCert [**?**] provides a centralized way to detect the log misbehavior which is again pruned to SPoF issue. LOCALPKI [140] was created for usage in the context of IoT, where the local authority is in charge of utilizing the public key to verify the user's identity. The certificates issued by the local authority are stored by a third party, which are trimmed to SPoF. In NBPKI [141] RA is in charge of authenticating the user's identification, and the NA maintains the user's status as trusted or untrusted based on the RA's decision. The malicious RA has the potential to compromise the system's integrity.

The $PA-PKI$ in article [142] uses *Practical Byzantine Fault Tolerance*($pBFT$) consensus model which allows a certain number of faulty nodes. If the number of faulty nodes exceeds that certain limits the whole network will be reset. Moreover, $pBFT$ consensus mechanism used in $PA-PKI$ of article [142] and $DB-PKI$ of article [150] is prone to the *Sybil attack*. The $Block-CAM$ in article [143] has used the consortium blockchain platform for developing their PKI. The major limitation of using this platform is making the entire system semi-centralized since the consensus is managed by a certain number of participating nodes. Thus, it deviates from the decentralization concept of blockchain. Our proposed model is completely decentralized to all existing nodes in the network. The transactions of $BCTRUST$ in [144], depend upon the degree of trustworthiness of a participating node. Once the node is declared as the trusted one, then every node in the same network has to consider that node as the same. Moreover, all the transactions made by that node are also considered as valid transactions which may cause integrity loss and many other cyber threats. In our model verification is done on every transaction where node identity already padded, it does not verify only such node based identity. This feature makes our model more secure compared to [144]. The implementation of PGP of both server and client-side participating nodes in $BlockPGP$ [145] causes heavy computational overhead which is the major drawback of such PGP based infrastructure.

In the case of $PB\ PKI$[146], the transactions is stopped if any anonymous node

requests to join the network. When new anonymous node requests to join the network, the entire network is disrupted until the joining request is processed. The developed PKI $TTA\ SC$ in article [147] suffers from the loss of control issue over the blockchain network if it loses the key pair of the lead node under some cyber attacks. The DDoS attack to a particular lead node can make the system destabilized. The $CERT\ CHAIN$ in article [148] uses *dependability rank* based consensus algorithm where the elected CA was responsible for increasing the trustworthy degree. Depending upon the degree of trustworthiness the node will be elected as CA. Thus, a DoS attack on the particular CA can cause damage in further transactions. It also uses the $PBFT$ consensus algorithm which may cause a Sybil attack. In the $CERT\ LEDGER$ of the article [149], the CA is responsible for publishing the revoked data after every transaction. Thus, a DoS attack on that particular CA can disrupt the entire network.

The developed PKI IKP in article [151] depends on the bitcoin's language script which becomes hard to implement. The transaction process in IKP depends on the trustworthiness of the CA and if the CA is misconfigured then all transactions within the network will be discarded. The $FLY\ CLIENT$ of article [152] does not use an authentication process to validate the participating node identification. So, the developed PKI is prone to $MITM$ attack. In $BLOCK\ QUICK$ of the article [153], the malicious block can only be detected by using the consensus group score. So, for a single malicious node, the whole branch will be discarded which will reduce the efficiency of the network.

4.6.2.3 Attack and Defense

The primary feature of the developed blockchain based PKI is the smart contract where the conditions such as the validity and threshold of the signer node are verified. The smart contract is solely responsible to allow the signer nodes to validate the key pair of requested nodes otherwise the nodes will be rejected. This feature avoids the DDoS and MITM attacks for the developed PKI. The proposed permissionless blockchain environment on the $GETH$ platform adopts the trust model of WoT where nodes are allowed to choose their own CA.

In the hierarchical trust model, the processing power required to calculate the NONCE is high, whereas WoT does not require any NONCE calculation. The NONCE calculation can prevent MITM attacks. However, we have avoided it intentionally in our PKI to make it lighter compared to existing literature. From the storage point of view, only the hash value of each node is considered for the on-chain storage and the entire data is considered for the off-chain storage. Different attacks addressed in the current blockchain based PKI are reflected in Table 4.6. Table 4.7 reflects the various attack resistance comparison of the proposed model in contrast to other existing blockchain PKI models.

Table 4.6: Different Threats & Its Defence

Attack	Basic Definition	Prevention Mechanism	Sustainability
DoS	The elected CA may initiate a huge number of transactions.	The proposed model defines a threshold for every node for becoming a CA and the implemented smart contract checks the given threshold with the RID. If the RID exceeds the threshold, the participation of CA will be rejected (see. Algo. 3).	Moderate
DDoS	Multiple elected CAs overload the network by initiating multiple transactions.	The nodes in the network can become a CA if the RID is less compare to the given threshold. (Algo. 3)	Moderate
MITM	An intermediate node may try to modify the transaction. This can be done in two ways such as modifying the content or modifying the sender/receiver node address.	Hash prevents content modification. The node verification process at the smart contact resolves the address violation part.	low
51%	It is an attack on blockchain where attackers acquire the control of more than 50% of the network's node address and cause faulty transactions.	Before initiating the transaction, the node identity will be checked and only the active node of the network will be allowed for the transaction. (Algo. 2)	low
Injection	Injecting multiple unknown nodes to access the data	The adopted WoT does not allow the joining of a random node in the network.	low
Routing	Tampering the data during the transaction	Hashing is used to secure the information.	low
Eclipse attack	The attacker may have a distributed botnets for replacing the actual node addresses by the false addresses.	For every transaction, the PID will be checked for availability and WoT model restricts random joining of nodes	low

Table 4.7: Attack resistance comparison

PKI	DoS	DDoS	MITM	51%	Injection	Routing	Eclipse
PA-PKI [142]	✗	✗	✓	✗	✗	✗	✗
Block-CAM [143]	✓	✓	✓	✗	✗	✗	✗
BC-TRUST [144]	✓	✓	✓	✗	✗	✗	✗
BLOCK-PGP [145]	✗	✗	✓	✓	✗	✗	✗
PB-PKI [146]	✓	✗	✓	✗	✗	✗	✗
TTA-SC [147]	✗	✗	✓	✓	✗	✗	✗
CERT-CHAIN [148]	✗	✗	✓	✗	✗	✗	✗
CERT-LEDGER [149]	✗	✗	✓	✗	✗	✗	✗
DB-PKI [150]	✗	✗	✓	✗	✗	✗	✗
IKP [151]	✗	✗	✓	✗	✗	✗	✗
FLY-CLIENT [152]	✗	✗	✓	✗	✗	✗	✗
BLOCKQUICK [153]	✗	✗	✓	✗	✗	✗	✓
Proposed System	✓	✓	✓	✓	✓	✓	✓

4.7 Conclusion

The proposed research work identifies several issues of conventional PKI and blockchain based PKI. In this regard, this work proposes a blockchain based PKI which is assisted by a smart contract and DPoS consensus algorithm. This work explores different existing solutions such as log based PKI, web of trust (WoT), and the blockchain based PKI system to deal with the various limitations and cyber threats of existing PKIs. The primary objective of this work is to create a blockchain based decentralized public key infrastructure which takes advantage of both the blockchain transparency and the web of trust model. The inclusion of smart contracts along with participant, signature and revoke modules in our work achieves the aforementioned features. The primary role of the adopted smart contract is used to validate the identity of the signer node and to check the threshold value for becoming CA.

The DPoS consensus algorithm used in our PKI reduces the timing complexity of the transactions which makes our PKI affordable for lightweight applications. The performance of the proposed PKI system is evaluated based on the latency of the key generation, key validation, and signature revocation process. The gas utilization on the Ethereum platform is minimal for the initialization process and transactions. The proposed PKI can prevent DoS, DDoS, MITM, 51%, Injection, Routing and Eclipse attacks. The developed smart contract used in our blockchain based PKI system is lighter to address the issue of storage limitation.

However, the issue found in the developed PKI is the computational overhead which is due to the CA selection process. As per the blockchain feature the CA needs to be selected for every transaction. This process no doubt removes the potential barrier of single point failure by introducing the decentralization concept. In a large size network, this process can increase the compuational overhead as the search space of CA selection is equals to the network size. Lowering this overhead, it can increase the network performance. With this as a primary objective this chapter aims to develop a clustering based PKI to minimize the search space. In addition, the developed PKI also considers the trust value for selecting an efficient node as the CA.

Chapter 5

Clustering and Trust enabled Blockchain-based Public Key Intrastructure

A blockchain-based PKI has been developed in Chapter 4 to deal with various Blockchain-based attacks including DoS, DDoS, MITM, 51%, Routing, Injection, and Eclipse attacks. In addition, the proposed PKI provides a fair chance for each peer to become the CA by deploying light smart contracts along with the DPoS consensus mechanism which makes it suitable for lightweight devices.

However, the issue found in the developed PKI is the computational overhead which is due to the CA selection process. As per the blockchain feature the CA needs to be selected for every transaction. This process no doubt removes the potential barrier of single-point failure by introducing the decentralization concept. However, a network may have a large number of transactions and participants. Selecting a CA for each transaction using PoS or PoA may cause a significant amount of block propagation delay, which can reduce network efficiency drastically. [1] This chapter proposes a different approach to increase the efficiency of Smart Contract assisted Blockchain-based PKI. The proposed approach creates clusters of participant nodes based on their validation time, response time, and trust. This method selects a cluster based on the budget of response time and validation time given by the node that intends to start a transaction. Thereafter, the node which has the highest trust in that cluster is chosen as a CA for the next transaction. Instead of searching on all participant nodes, our approach searches on the nodes of the chosen cluster which reduces the searching space of the CA selection process. This research work adopts a trust evaluation approach where the trust factor is quantified based on its experience and reputation. The node trust is reevaluated after every successful and unsuccessful transaction. A node that performs more successful transactions has more

[1] The outcomes of this chapter was published in “MLCSS-2022"[C2], and “IEEE ACCESS" [J3]

trust value. The node that has a higher trust value has a higher probability to be selected as a CA for a transaction.

5.1 Introduction

Communication via an unprotected network can only be guaranteed by the verification of each participant's identity. For example, a man-in-the-middle (MITM) attack [157] may be used to intercept communication and imitate the participant's involvement. Public Key cryptography [158] is one of the promising solutions to secure communication in an untrusted network. Since the introduction of public key cryptography, the verification of the trustworthiness of a participant's public key has been a prominent issue. In this context, "trusted" means that the private key is known only to the intended communication partner. If both participants involved in the communication know the same secret, such as a password, the problem becomes simplified significantly. Sharing a private key in a large-scale network is not always possible. So Public Key Infrastructure (PKI) [159] can be used as an alternative solution for public key cryptography.

The use of encrypted communication protocols is being actively pushed and supported more than it has ever been before. Regular HTTP connections, which appeared to be fair in the past [160], are now described as "not secure," whereas HTTPS connections are unmistakably labeled as "secure." This change in the appearance of security indicators in the address bar has been implemented by browser vendors such as Google Chrome [161], Mozilla Firefox [162], and other popular browsers. Cryptographically protected protocols such as Transport Layer Security (TLS) protocol are becoming the usual solutions as more administrators and developers become aware of the risks associated with using insecure protocols [163]. The risks associated with insecure protocols are increasing as more people become aware of them. The TLS PKI also has a vulnerability called the weakest-link security problem, which means that any trustworthy CA may create a valid certificate on its own for any domain name. A client will regard a certification authority to be trustworthy if that authority's certificate is included in the client's list of root CAs or if the certificate was signed by another trusted CA. Both $X.509$ [164] and PGP [165] are the other two widely used protocols for securing internet-based communication.

An attacker may undermine the integrity of the system as a whole by gaining control of a single root or intermediate CAs. To overcome the issues caused by the generalized web-based security protocols there are two different solutions present Log-based and Web of trust. Among all solutions, $CertificateTransparency(CT)$ framework [166] by Google is the most popular one. It makes certificates publicly accessible by using append-only logs for updating and maintaining the list of log servers. Even if the contents of the log may be read and shown to be consistent, log servers have the option

to disregard any requests that are sent their way. Last but not least, a gossip protocol is required in order to prevent a split-world attack [167], which occurs when a malicious log server presents various clients with conflicting copies of the log. Therefore, in order to accept a malicious or compromised CA, each certificate issuance should include numerous CAs, and all activities should be documented in a safe and completely dispersed manner. This is necessary in order to tolerate the presence of a CA. The other $Log-basedPKI$ solutions include $AccountableKeyInfrastructure(AKI)$ [138], $AttackResilienceKeyInfrastructure(ARPKI)$ [139], etc. The main issue the log-based PKI is facing is the centralized $IntegratedLogServer(ILS)$. The presence of ILS makes the Log-based PKI solution prone to a single-point failure. The third-party can easily get access to the ILS server by means of which the entire system will fail to maintain the integrity level. Another possible solution to the conventional PKI system is the $WebofTrust(WoT)$ based PKI. The $WebofTrust(WoT)$ includes notary-based solutions such as $LocalPKI$ [[140] and $Notary-basedPKI$ [141] that are intended to offer different PKI systems that enable the end-user to use their known trusted node to act as the CA. In this type of PKI, the $NotaryAuthorities(NAs)$ replaced CAs to store the only signed hash of the certificate and its serial number in the database. However, with $notary-basedPKI$ and $LocalPKI$ systems, users and NAs must have confidence in order to oversee the functioning of certificates. Therefore, it is necessary to prevent notaries from certifying bogus certificates and signatures.

To overcome the lacuna present in $Log-basedPKI$ and WoT, blockchain-based PKI becomes an emerging solution. The characteristics such as immutability, transparency, security, and distributed ledger are the technical benefits of blockchain which make it a more appropriate technique for internet-based communication. A promising characteristic known as decentralization of internet services is the key concept presented behind blockchain technology. Instead of depending on a single CA for issuing the certificate, this technique enables the network to have multiple CAs for different communications. Adopting multiple CAs simply avoids the single-point failure limitation of conventional PKI systems.

5.2 Related Work

This work is mainly motivated by 3 major aspects (i) Trust calculation, (ii)Clustering of participant nodes to reduce the searching space Validator, and finally (iii) PKIs. In section 5.2.1, existing literature on the trust calculation of node in a Point to Point (P2P) network with and without blockchain are discussed. In the section 5.2.2 different blockchain-based clustering mechanisms are discussed where machine learning plays a crucial role. In section 5.2.3 various blockchain-based PKI systems are discussed.

5.2.1 P2P network and Blockchain network trust calculation

In this section, various trust calculation methods in P2P network (section 5.2.1.1 and Table 5.1) along with the blockchain network node trust calculation methods (section 5.2.1.2 and Table 5.2) are reported.

5.2.1.1 P2P network trust calculation

The Bayesian network trust model introduced by Wang et al. [168] employs the Bayesian network to compute the trust degree and the probability technique to determine the node trust value, which subtly increases algorithmic complexity. The trust parameters are quantified into the [-1,1] range, which may be stated intuitively as a full trust to total untrust node. A model for calculating trust based on evidence theory was proposed by Yu et al. [169]. Evidence of a node's support has been used to recognize that particular node as the target node.

A distributed trust calculation model called PageRank was suggested by Yamamoto et al. [170]. This model estimates the trust value of nodes by using the PageRank algorithm that is shared throughout the network. PeerTrust, which was developed by Xiong et al., makes use of many factors to automatically alter the trust value of nodes over time, ultimately selecting the high-trust node as the one with which to connect [171]. PeerTrust determines the trustworthiness of a node by taking into account a number of criteria relating to a transaction and the environment of the network. Based on the D-S evidence theory, Wen et al. [172] suggested a way to identify trust relationships and confidence intervals between peers. In order to determine the reliability of the nodes, the model makes use of both the arithmetic average and the Bayesian approach simultaneously.

Song [173] presented a model for the trust that makes use of fuzzy logic inference to calculate the local trust value of a peer and aggregates the recommendation information. The principles for logical reasoning using linguistic trust metrics are provided by fuzzy logic. For DHT-based P2P networks, the PowerTrust system [174] was suggested, which makes use of the Power-law distribution of peer feedback. Using a distributed ranking method, PowerTrust dynamically chooses a limited number of power nodes that are the most trustworthy. PowerTrust dramatically increases global reputation accuracy and aggregation speed by using a look-ahead random walk approach with the power nodes.

5.2.1.2 Blockchain network trust Model

Sun et al. in [175] proposed a trust calculation model for a blockchain network that calculates the trust value of a node by acquiring the working state and behavioral information of that intended node. The final trust is calculated by aggregating the trust value generated during the transaction and the trust value generated by the behavior.

Table 5.1: Related work based on P2P network trust calculation

Ref	Trust calculation model	Network type	Blockchain Implementation
[168]	Baysian Network	Peer-to-Peer	✗
[169]	Evidence Theory	Peer-to-Peer	✗
[170]	Page Rank Algorithm	Peer-to-Peer	✗
[171]	Successful transaction	Peer-to-Peer	✗
[172]	Evidence Theory	Peer-to-Peer	✗
[173]	Fuzzy logic inference	Peer-to-Peer	✗
[174]	Node Feedback	Peer-to-Peer	✗

For a blockchain-based online payment system, Ahn et al. in [176] suggested a methodology for estimating trust and reputation using the values contained on a blockchain ledger. Information from ratings and transaction histories has been effectively utilized to calculate reputation and trust levels. The blockchain-based payment system keeps track of its entire history. While regularly validating and confirming such values in the background for dependability without impairing user experience, the model uses a small cache of key data to speed up searches.

She et al. in [177] proposed a blockchain-based trust model for detecting the malicious node in the case of the wireless sensor node. For calculating the trust value four different attributes including the node behavior, response time, transmission delay and forwarding rate have been considered. The state of the node is further divided into two different groups such as working or non-working state. Initially, a node has been verified for its state and if the state is found working then only the other three parameters are considered otherwise the node will be discarded from the network. The final trust has been calculated by aggregating the delay factor, forwarding rate, and response time.

Zhao et al. in [178] presented a model Trustblock to calculate the trust of the data layer devices for the Software Defined Network (SDN). Direct, Indirect, and Historical trust are the three key parameters considered for calculating the final trust of a node. The final comprehensive trust is calculated by normalizing the three different types of trust with three different weight factors $w1$, $w2$, and $w3$. These weights are calculated by using the entropy value.

Inedjaren et al. in [179] have proposed a blockchain-based distributed framework for calculating the trust in the Vehicular Adhoc Network (VANET). The node uses two types of control messages such as $HELLO$ and Traffic Control (TC) through the OLSR routing

protocol for any kind of communication. The trust of the node is calculated by using the membership value of each control message which can be of *verylow*, *low*, *medium*, *large*, and *verylarge*. Next, the defuzzification rule is applied to the membership value of $HELLO$ and TC to obtain the final trust value of that particular node.

Table 5.2: Related work based on Blockchain trust model

Ref	Key Feature	Blockchain Platform	Trust Usage	
			PKI	CA Selection
[175]	Behavioral information	✓	✗	✗
[176]	Transaction History	✓	✗	✗
[177]	Node behaviour, response time, transmission delay, data forwarding rate	✓	✗	✗
[178]	Transaction history and Peer feedback	✓	✗	✗
[179]	Defuzzification	✓	✗	✗

5.2.2 Blockchain clustering

In the current section, different machine learning-based clustering approaches for blockchain networks are discussed. Table 5.3 shows the summarization of the considered literatures.

Zola et al. [180] proposed a machine learning-based method for detecting malicious activities in a bitcoin network. Initially, the clustering algorithm has been applied in order to make different clusters of malicious and non-malicious data present in the blockchain. Finally, different ensemble machine learning by using different classification algorithms such as Random Forest, Adaboost, and Gradient Boosting for classification purposes. The proposed model shows a 99.68% accuracy level.

Chawathe et al [181] proposed a novel approach for clustering the bitcoin data for behavioral analysis. For clustering, the K-Means clustering algorithm has been considered. Mahalanobis distance metrics have been used in order to evaluate the identified clusters.

Huang et al. [182] proposed a novel approach as the Behavior Pattern Clustering (BPA) algorithm which takes the blockchain transactional data over time as the input. The proposed algorithm has been evaluated by considering 1321 numbers of records as the nodes. BPA has been compared with the K-Means and K-Means ++ algorithms to show its efficiency.

Ermilov et al. in [183] reported an approach to identify the blockchain data owner.

This can e performed by using behavioral pattern analysis and off-chain data available publicly. For behavioral pattern analysis, machine learning clustering algorithms have been applied. Web crawling and manual analysis of various bitcoin data providers are used for the off-chain data analysis.

Harrigan et al. [184] have used the machine learning clustering algorithm for making different clusters of the available bitcoin data available up to February 2016. The clustering method has been implemented to the publicly available data to identify fraudulent transactions. As a result, the author has created a supercluster of the identified attacks.

Fleder et al. [185] reported a novel approach by using the machine learning clustering algorithm to identify the known and unknown users. For empirical analysis, the raw bitcoin data up to December 2013 has been considered.

Li et al. in [11] a blockchain partitioning technique based on node community clustering In a CPS, it is a way for grouping nodes into distinct groups. Each group is a chain, with some nodes sharing the same information. Depending on the approach, a node can be added to several chains. The nodes on the same chain only need to synchronize the data of the nodes that have joined the chain for data synchronization. The proposed technique results in reduced cross-link communication data lower system network strain and improves system communication efficiency. At the same time, it decreases the amount of data that nodes store that isn't relevant and speeds up data querying.

Table 5.3: Related work based on Blockchain clustering

Ref	Clustering Technique	Key Feature	Blockchain clustering criteria	
			Data clustering	Node clustering
[180]	K-Means	To detect the malicious activity in the bitcoin network.	✓	✗
[181]	K-Means	For node behavioral Analysis.	✓	✗
[182]	BPA	For behavior pattern analysis.	✓	✗
[183]	K-Means	To identify the blockchain data owner.	✓	✗
[184]	K-Means	To detect the fraud transaction.	✓	✗
[185]	AHC	To detect the known and unknown users.	✓	✗

5.2.3 Blockchain Based PKI

Abba et al. in [186] proposed a blockchain-based PKI $BB-PKI$ for managing the certificates. In this, a client initially requests a certificate from the registering authority (RA), and then the RA forwards the request message to the corresponding CA for certificate issuance. Within the network, there are multiple CAs and RAs. The main objective of this work is to avoid the single point of failure (SPoF). Lukasz et al. in [187] proposed a blockchain-based PKI known as $BlockPKI$. The main objective of this model is to automate the certificate issuance system. The domain owner defines the number of CAs who can issue and validate the certificate. Upon receiving the request from the node for a certificate the smart contract will be invoked and among the defined CA depending upon the availability, one CA issue and validate the certificate.

Yakubov et al. in [188] proposed a $blockchain-basedPKImanagementframework$ with the objective to avoid the SPoF limitation of the traditional PKI system. In the developed PKI each CA contains its own smart contract dealing with all relevant information regarding the certificates including the hash of previously issued or revoked certificates. Bo et al in [189] proposed a PKI framework $Cecoin$ for bitcoin. For issuing the certificate the PoW consensus mechanism is being used. The participating node will try to solve the puzzle or NONCE issued by the initiator. The node solving the puzzle first issues the certificate for the transaction. Tewari et al. in [190] proposed $X.509Cloud$ as the blockchain-based PKI system. The main idea present in this work is to issue different certificates for new requests and the certificate revocation process.

5.3 Problem Statement and Motivation

Evidently, the decentralization characteristic eliminates the limitations inherent in the conventional centralized PKI system. In a blockchain network, every transaction requires the selection of a CA. Therefore, a large number of transactions need extensive computing effort. This CA selection procedure becomes the major cause of network computation overhead, which reduces the network's performance. To circumvent the problem, this network clustering is a potential solution.

In addition, the blockchain nodes perform the transaction with the other participant nodes with the presence of some participant node called as CA. In this node interaction process, trust is the key factor. The so-called “don't trust" issue of blockchain considers a poor relationship among all nodes. Even though BC-PKI has transparency, decentralization, immutability, and security still it faces a credibility crisis. A credibility crisis explains a scenario of whether the participant nodes are creditable or not for a successful transaction. Choosing a node as a CA which performs more number of successful transactions will increase the trust of that node and the efficiency of the PKI as well. Hence trust

value can be one of the most important parameters for the BC-PKI network. Therefore, the current work considers the clustering of participant nodes of the blockchain network and trust value calculation as the most inclusive factors.

5.4 Objective

The objective of this work us categorized into two subsections as follows:

- **Approach 1** The objective of this work is to implement K-Means clustering to make different clusters of the entire network. The contributions of this research work can be summarized as follow:
  - To implement the K-Means clustering technique to make different cluster of the entire network.
  - To measure the effectiveness of the clustering on the blockchain performance in terms of computational time.
- **Approach 2** It is the extended version of **Approach 1** with the primary focus to reduce the CA search space. In addition, the proposed Clustering and Trust enabled Blockchaon-based PKI ($CTB-PKI$) adopts the trust based mechanism to select an efficient node as the CA. The key contribution of this work can be summerized as follows:
  - The proposed $CTB-PKI$ implements a cluster-based CA selection approach which reduces search spaces significantly. The clustering algorithms used in our $CTB-PKI$ CA selection process are based on 3 parameters: trust, response time, and validation time. The proposed $CTB-PKI$ uses the $K-Means$ with $silhouettescore$ and $DBScan$ clustering algorithms.
  - The proposed $CTB-PKI$ quantified the trust value-based experience and reputation of the participant node. The reputation is based on direct and indirect trust and the experience is calculated based on the number of successful and unsuccessful past transactions
  - The proposed PKI is evaluated based on the three metrics (i) response time with and without clustering, (ii)validation time with and without clustering, and (iii) Gas cost used for different transactions. The reduced latency of the proposed $CTB-PKI$ makes it suitable for Blockchain 2.0 and 3.0 application domains.

5.5 Machine Learning based Clustering and its need in Blockchain

Clustering is an unsupervised machine learning technique for making multiple groups based on some similar features [191]. In the case of blockchain, the clustering techniques can be used two different scenarios.

- To make the clusters of blockchain data to identify the malicious activity that occurred in the network.
- To make clusters of the blockchain node to minimize the search space while selecting a CA for one transaction.

Clustering the blockchain data is one of the most popular use cases of machine learning-based clustering technique which can be reflected in section 2 and Table 5.3. The second use case of clustering remains unexplored. A Blockchain network contains multiple number nodes and also multiple transactions. As per the blockchain feature, every transaction must have a different CA for validating the transaction. Searching different CA every time in the entire blockchain network will take numerous times which can also increase the network overhead. So, clustering the blockchain network emerges as the solution to decrease the network overhead by limiting the search space of CA selection. The main issue that node clustering faces are finding the appropriate features for grouping the nodes into different clusters. For this, the response time and validation time can be the two parameters for making the clusters.

5.6 Methodology

The proposed clustering based PKIs are implemented in the open-source Ethereum platform ($GETH$). The *solidity* v 0.4.24 scripting language and $Truffle\ Suit$ are used to deploy the smart contract to the blockchain environment. A system with Windows 10 OS, 8GB RAM, Intel i5 with 2.8 GHz clock speed, 1TB HDD, and 500GB SSD is used to implement the proposed blockchain-based PKI. Section 5.6.1 and 5.6.2 show the methodology description for Approach 1 and Approach 2 respectively.

5.6.1 Methodology for Approach 1

The main objective of this research work is to introduce the clustering technique to divide the entire blockchain network into various groups. The main concept present in this research work is to construct clusters based on the response time of the node. Initially, all nodes are considered as a single group. After every transaction the response time factor

(RTF) will be calculated which becomes the primary factor for making the clusters. In the current research work the K-Means clustering technique along with the elbow method as the internal validation measure. Depending upon the RTF the node can be a part of a particular cluster.

5.6.1.1 Working Principle

When a node or miner wants to communicate then it has to specify the threshold RTF needed for completion of the transaction. Each cluster is identified by a value which is the average RTF of all present nodes. When a miner node wants to communicate or to make a transaction then it has to select a group of nodes for becoming the CA of that corresponding truncation. It can be achieved by implementing the Proof of Authority (PoA) consensus model. The PoA consensus model selects a particular node from the cluster based on its trust factor (TF). The TF is just a counter which is increased by 1 for every successful transaction validated by that particular transaction and reduced by 1 for each unsuccessful transaction. After selecting the particular node from the group, the smart contract is revoked to check the validity of the node for becoming the CA and to validate the transaction from every present node. Figure 5.1 shows the workflow of the proposed work. The pseudocode of the proposed work is provided in algorithm 6.

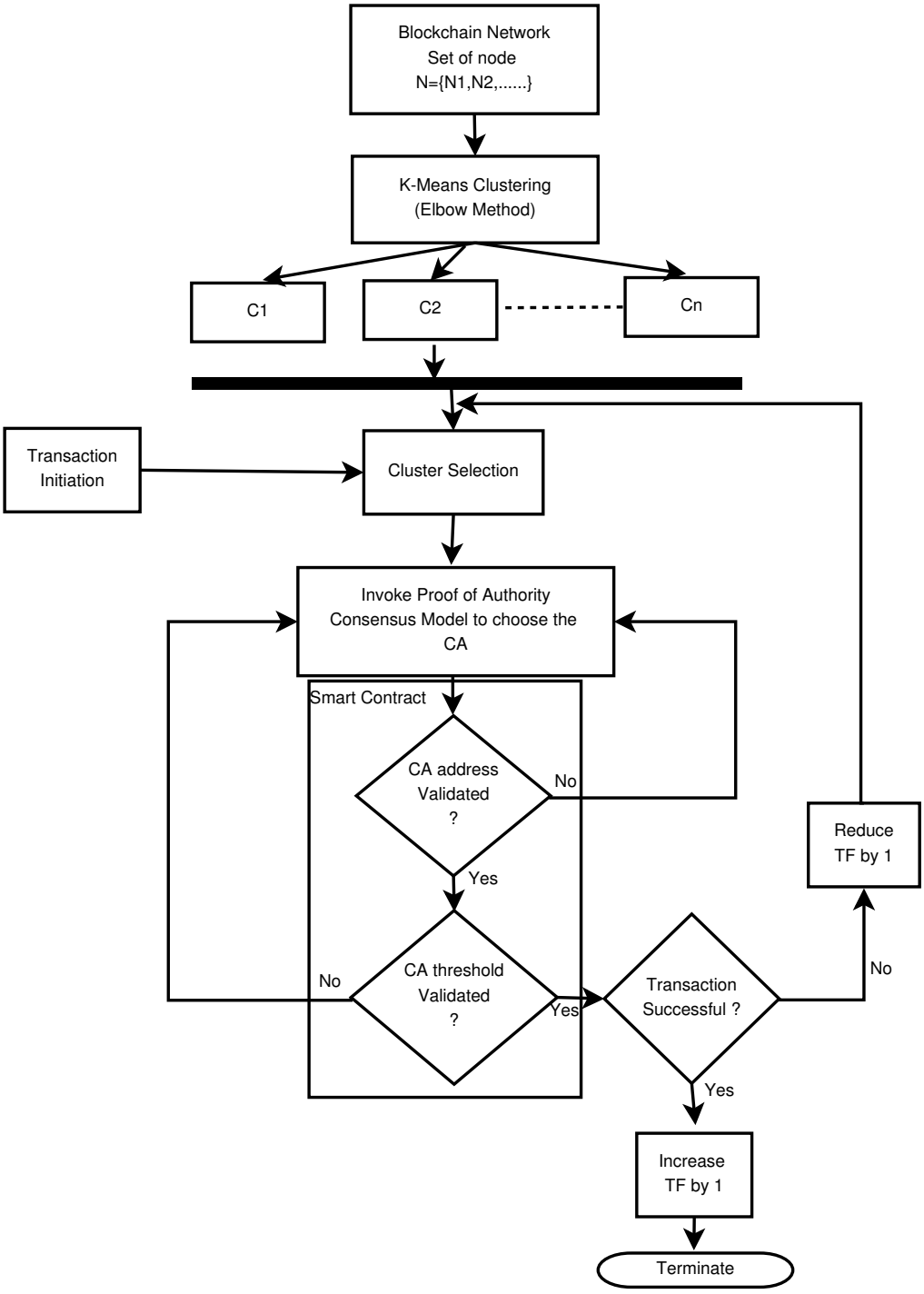

Figure 5.1: Workflow of the proposed error correcting models

Algorithm 6 Algorithm for the proposed work

REQUIRE: **Set of Nodes N=[**N_1,N_1,N_1,.....,N_n**],** NA_i**: Node Address (NA) of a node,** $N_{Threshold}$**: Limit of a node N for becoming CA**
PROC $K - Means()$
Implement $Elbow()$ to determine the number of clusters (K)
Transaction Initiation
Select the required cluster for selecting the CA
Invoke PoA() Consensus
Select one cluster for getting the CA
Select CA
PROC $SmartContract()$
if (NA_{CA} == valid) **then**
    Verify the $CA_{Threshold}$
    if ($CA_{limit} > CA_{Threshold}$) **then**
        Process the transaction
    else
        Abort the transaction and return to 6
    end if
else
    Abort the transaction and return to 6
end if
if (Transaction==Successful) **then**
    Increare the Trust Factor TF by 1 and terminate
else
    Reduce Trust Factor TF by 1 and return to 4
end if

5.6.1.2 K-Means Clustering Algorithm

The K-Means technique is a benchmark method for splitting the entire dataset or nodes into K non-overlapping subgroups, each including multiple cluster members. In the current research work, the RTF has been considered as the key factor for making different clusters. K-Means technique provides a method to make the clusters but the main issue is deciding the number of K. To solve this issue there are several methods present out of which the Elbow method is a well-known approach.

An elbow approach is used to graphically demonstrate the validity of the number of clusters. The sum of squared errors (SSE) is determined for each k. A chart depicting an arm with an 'elbow' is used to denote the best k. In general, the SSE approaches zero increasing the number of clusters by a factor of k. SSE is calculated by using the equation 5.1 with RFT as the Response Time Factor and VT as the key verification time and N is the total number of nodes present in the blockchain network.

$$SSE = \sum_{n=1}^{N} (RTF_n - VT_n) \tag{5.1}$$

5.6.1.3 Proof of Authority (PoA) Consensus Model

It depends upon the reputation of a participating node. The model first verifies the identity and the behavior of the participating node before making it a lead node. Once the transaction will be completed successfully the reputation of the node will be increased by one which will help it in becoming a lead node in the further transaction. If the communication is a failure then the degree of reputation will be decreased. The node having the highest degree of reputation will be chosen for validating the key pair and transaction within the network.

5.6.2 Methodology for Approach 2

his section focuses on the different building blocks of the proposed CTB-PKI system. The proposed CTB-PKI is implemented in the open-source blockchain platform Go Ethereum ($GETH$) with the smart contract as a key element. Figure 5.2 reflects the block structure of the proposed work. Initially, the node that wants to initiate a transaction has to go for the CA selection. For selecting the appropriate CA the node needs to select the cluster first.

Clustering is a machine learning-based technique that allows making different groups of data points having similar characteristics. The primary objective of the clustering technique is to make an intrinsic grouping of an unlabelled dataset. The main question present behind this technique is how to define the number of clusters. To solve this problem various algorithms including K-Means, K-Means++, DBSCAN, Agglomerative Hierarchical Clustering, etc are present. For the current study, the K-Means (section 5.6.2.2) and DBSCAN methods are being used to determine the number clusters.

The cluster selection depends on the minimum response time from the transaction initiator node. From the selected cluster a node will be selected as the CA depending upon the trust value it has. A node having a higher trust value will have a higher probability to become a CA. After selecting the CA the certificate is issued to the requesting node and also forwarded the same for the network for synchronizing the same in the $DistributedLedger$ (DLT).

5.6.2.1 Model Description

The proposed CTB-PKI consists of different modules such as the Participant, Validation, and Signature Revocation.

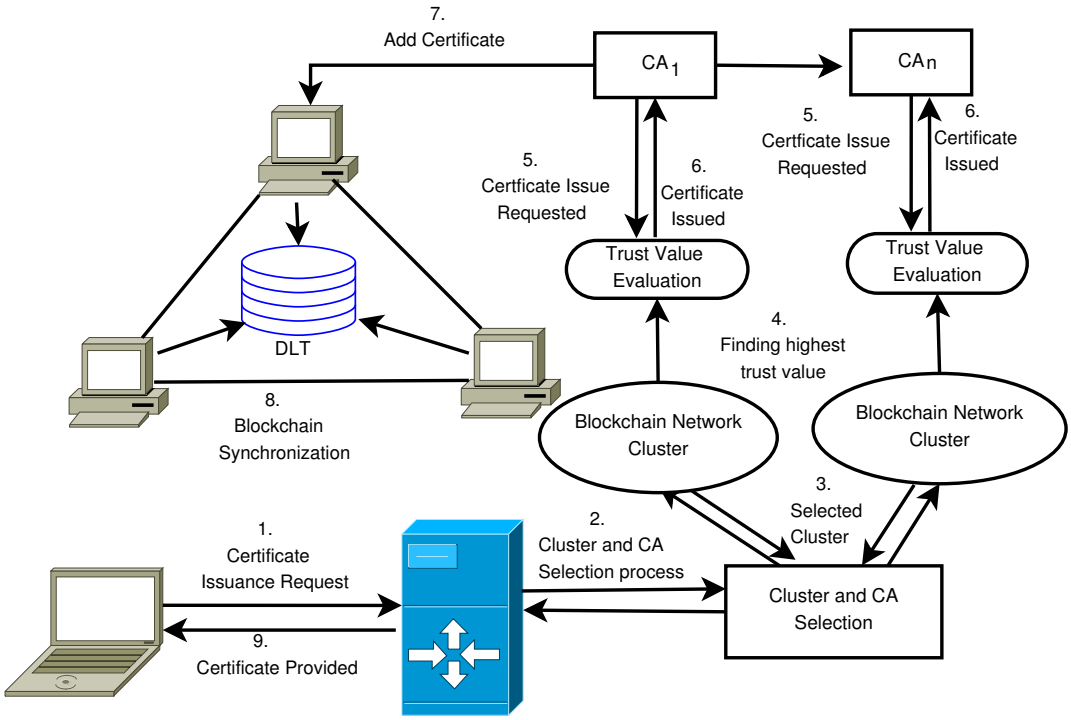

Figure 5.2: Block diagram of the proposed CTB-PKI system

1. **New Participant:** This module is called when a node wants to communicate in the blockchain. Before communicating the node status in the network is verified. If the node is found to be a new joinee to the network then the following parameters are invoked. This module has a 7-output tuple for a node when invoked as shown in equation **??**.

$$NewParticipant \leftarrow T\{Node_{id}, N_{ETHaddress}, Pr - Key_N, Pu - Key_N, N_{Expiry}, N_{Rev}\} \tag{5.2}$$

   - $Node_{id}$: It is a random number provided to identify a particular node in the blockchain network.
   - $N_{ETHaddress}$: It is an unique $EThereumaddress$ provided by the $GETH$ environment to a node N.
   - $Pr - Key_N$,$Pu - Key_N$: The $Pr - Key_N$ and $Pu - Key_N$ are the private and public keys of a Node (N) to be used during the communication.
   - N_{Expiry}: It is the maximum or threshold limit for a node i for which the node N can become a CA.
   - N_{Rev}: It is a counter of the node N to indicate the number of times a node becomes a CA. With the initialization, this counter value is set to 0.

2. **Validation:** In this section the validation will be done for two different nodes such as the transaction initiator and selected validator. For instance, node A wants to

initiate a transaction with selected CA as B. Then, the input tuple of this module is reflected in equation 5.3.

$$Validation \leftarrow T\{A_{id}, B_{id}, Exp_B\} \tag{5.3}$$

A_{id} and B_{id} are verified in or of both transaction initiator and the CA are checked for their existence in the network. Exp_A is another input to this module for verifying the eligibility of a node as CA. Both of the conditions are executed in a smart contract. If conditions are satisfied then the node will be allowed to have the corresponding CA for validating the transaction.

3. **Signature Revocation:** After every transaction this module will be invoked. Taking the previous instance with A as the initiator and B as the CA into consideration the input tuple of this module is shown in equation 5.4.

$$SignatureRevocation \leftarrow T\{Rev_B, B_{id}\} \tag{5.4}$$

After every transaction, the revocation id Rev_{id} of the CA will be incremented by 1. Every time the CA's Rev_{id} will be verified against the Exp_i to check the maximum limit of that node for becoming the CA.

5.6.2.2 K-Means Clustering

K-Means is one of the most popular clustering techniques [192]. The objective of the technique is to find the number of k clusters out of N number of data points. The performance of this algorithm depends upon the optimal k-value selection which is one of the biggest issues of this algorithm. To solve this issue there are several internal validation methods present such as the Elbow Method, Silhouette Coefficient, and Calinski-Harabasz [193]. For the proposed work the Silhouette Coefficient (SC) approach is adopted. To calculate the SC the two attributes response time (RT)and validation time (VT) of the blockchain node are considered. The SC can be calculated by using equation 5.5. Algorithm 7 shows the pseudocode of the K-Means algorithm.

$$SC = \frac{RT_i - VT_i}{max(RT_i, VT_i)} \tag{5.5}$$

5.6.2.3 DBSCAN:

Density-Based Spatial Clustering of application with Noise or DBSCAN technique is used to identify different clusters of the data points that are closed to each other depending

Algorithm 7 K-Means Clustering based on$\langle RT_i, VT_i, T_i \rangle$

Input: Set of Nodes N=[N_1,N_2,.....,N_n**]**
Output:Optimal k value
$k - max \leftarrow 18$
$k \leftarrow 1$
while k $\leq$ k-max **do**
    $CalculateSC$
    $printSC$
    $k \leftarrow k + 1$
end while
Obtain the optimal k with maximum SC value

on some measurement [194]. It has two inputs mpts and epsilon. Algorithm 8 shows the pseudocode for the DBSCAN method.

- mpts: It is the minimum number of data points required to form a dense region.
- Epsilon(ϵ): It is the distance measurement that is used to locate the next data points from any random datapoint.

Algorithm 8 DBSCAN algorithm based on $\langle RT_i, VT_i, T_i \rangle$

Input:N, ϵ, mpts
Output:Optimal k value
Cluster $\leftarrow \phi$
for $\forall$ n $\in$ N **do**
    mark n as visited
    X $\leftarrow$ GETNEIGHBOUR(n,ϵ)
    if ($|X|$ <mpts) **then**
        mark n as the noise
    else
        Cluster$\leftarrow$ Cluster $\cup$ n
    end if
end for

5.6.2.4 Trust Calculation

Trust is the value that plays a vital role in selecting a particular node as the CA for a transaction [195][196]. The trust (T) of a node can be calculated by two factors including (i) experience level (E) (section 1) (ii) reputation factor (R) (section 2). Notations used for calculating the trust value are reported in Table 5.4. The trust value of every participating node is calculated by using the equation 5.6 and 5.7 with w_R and w_E as the weight factors such as $w_R + w_E = 1$. Our work considers equal priority on the weightage of experience

and reputation parameters.

$$Trust = w_E \times E + w_R \times R \tag{5.6}$$

$$Trust = \frac{1}{2} \times E + \frac{1}{2} \times R \tag{5.7}$$

Table 5.4: Notations for trust calculation

Notation	Definition
E_0	Initial Experience level of a new joinee node
E_{min}	The minimum experience level of a node which is set to 0. The experience level is normalized between 0 and 1
E_{max}	Maximum experience level of node
R_{min}	The minimum experience level of a node which is set to 0
α	Feedback score after successful transaction
β	Feedback score after each unsuccessful transaction
E_t	Current experience level of a node n
E_{t+1}	Updated experience level
S_T	Number of Successful transactions
U_T	Number of Unsuccessful Transaction
$S_{T(C1-C2)}$	Successful transaction from cluster C1 to C2
$U_{T(C1-C2)}$	Unsuccessful transaction from cluster C1 to C2

1. **Experience Level** The experience level (E) is calculated by using positive experience (E_{pos}), and negative experience (E_{neg}). The T_{pos} and T_{neg} are responsible for increasing and decreasing the trust value of a node respectively. The experience level of a node will be updated after every transaction.

   (a) **Positive Experience:** For a transaction a node n acts as the CA. After the successful transaction the positive experience value (E_{pos}) follows the following linear equation 5.8.

   $$E_{t+1} = E_t + \alpha\Delta \tag{5.8}$$

   where Δ can be defined as the equation 5.9 with η as the value to normalize the experience value between 0 and 1.

   $$\Delta = \eta \times (1 - E_t) \tag{5.9}$$

(b) **Negative Experience:** A node n acts as the CA for a transaction. After every unsuccessful transaction the negative experience value ((E_{neg})) follows the equation 5.10.

$$E_{t+1} = Maximum(E_{min}, E_t - \beta) \tag{5.10}$$

2. **Reputation Factor calculation** The reputation factor (R) is the aggregation of intra-cluster trust and inter-cluster trust. In the proposed work network clustering is performed. For a transaction, the CA and the node n can belongs to the same cluster or a different cluster. If both of the nodes belong to a single cluster then the trust is called a direct trust (T_D) otherwise the trust is known as the Indirect trust (T_{ID}). T_D can be calculated as the equation 5.11:

$$T_D = \begin{cases} Maximum(R_{min}, \dfrac{S_T - U_T}{S_T + U_T}) & \text{if } S_T, U_T \neq 0 \\ 0, & \text{Otherwise} \end{cases} \tag{5.11}$$

For instance a node i of cluster $C1$ selects a node j as the CA of another cluster $C2$, then the T_{ID} of the node the CA is calculated as equation 5.12.

$$T_{ID} = \begin{cases} Maximum(R_{min}, \dfrac{S_{T(C1-C2)} - U_{T(C1-C2)}}{S_{T(C1-C2)} + U_{T(C1-C2)}}), \text{if } S_{T(C1-C2)}, U_{T(C1-C2)} \neq 0 \\ 0, \qquad \text{Otherwise} \end{cases} \tag{5.12}$$

5.6.2.5 Consensus Model

For the proposed $CTB-PKI\ Proof of Authority(PoA)$ consensus methodology is adopted. The key concept present behind this consensus method is to choose the CA depending on reputation or trust value. For every successful transaction, the trust value is updated as per section 5.6.2.4. The reputation of a node as CA will increase for every successful transaction and decrease for every unsuccessful transaction.

5.6.2.6 Blockstructure

Block is the key element in the blockchain. It is composed of two different components as block header and a body. The block header consists of (i) the hash of the previous block (ii) the time stamp at which the block is created (iii) NONCE which is the optional part that is kept only for the transaction using Proof of Work (PoW) and (iv) the Merkle root which is hash of the root of the Merkle tree. By storing the hash of the previous block

the chain of blocks is created which ensures the data integrity. A small change in the transactional data will be reflected as it significantly changes the Merkle root. This also simplifies the transaction verification process by only comparing the generated root hash of the Merkle tree with the stored one. The body of the block indicates the transactional data. Figure 5.3 shows the block structure used for the current work.

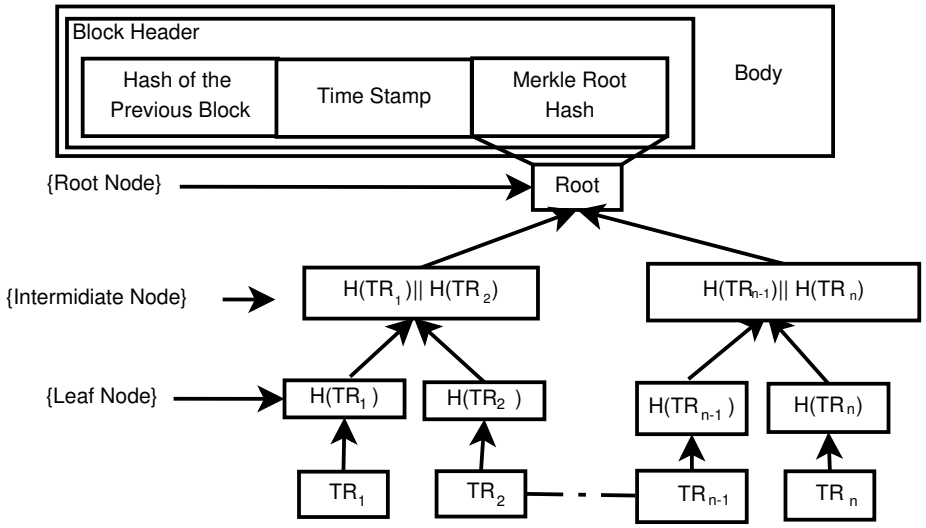

Figure 5.3: Blockstructure of the proposed CTB-PKI system

5.6.2.7 Working Principle

The core functionality behind the proposed $CTB-PKI$ is to select the CA for a transaction depending on the node trust value. The higher trust value enhances the probability of a node becoming CA. CTB-PKI method suggests an approach for calculating the trust of the nodes (see section 5.6.2.4). The decentralization characteristic enables the network to have different CA for different transactions. The blockchain network can contain a large number of nodes. So the search space for selecting CA every time increases the computational overhead. To avoid this issue, the proposed work adopts different clustering algorithms such as $K-Means$ and $DBSCAN$ to make different clusters of nodes (see section 2). Algorithm 9 and Figure 5.4 show the pseudocode and workflow of the proposed work. The working process of the proposed work is elaborated in the following steps.

Step-1

Initially, the clustering of nodes is executed depending upon two parameters such as $\langle RT, VT \rangle$. It is because the trust value of the participating node is set to 0 initially. The CA selection process can be done based on the input budget $\langle RT, VT \rangle$ by the participant node.

Step-2

After a certain number of transactions, the nodes are re-evaluated for the cluster with an input of 3 values $\langle RT, VT, T\rangle$. Each cluster have average RT and VT value named as RT_{avg} and VT_{avg}. For initiating a transaction the participating node provides a budget of response time RT_{budget} and a budget of validation time VT_{budget}. The cluster which has the least RT and VT compared to RT_{budget} and VT_{budget} is selected as the preferred cluster for our CA selection process. Thereafter all nodes of the selected cluster evaluate their rank by the equation 5.13.

$$Rank = W_R \times (1 - \frac{RT}{RT_{max}}) + W_V \times (1 - \frac{VT}{VT_{max}}) + W_T \times \frac{T}{T_{max}} \tag{5.13}$$

Equation 5.13 has three weighted parameters WR , W_V and W_T which indicate the priority of response time, validation time and trust respectively where $W_R + W_V + W_T = 1$(normalized). The equal priority mode means W_R=W_V=$W_T = \frac{1}{3}$. In general applications, trust and delay are considered fundamental parameters where the delay is $R_T + V_T$. In this sense the equal priority means W_R+W_V=$\frac{1}{2}$ and $W_T = \frac{1}{2}$. The Single priority mode means any one of W_R, W_V, and W_T is unity and the other two are zero; for response time priority, $W_R = 1$, $W_V = 0$, $W_T = 0$; for validation time priority, $W_R = 0$, $W_V = 1$, $W_T = 0$; and for trust priority, $W_R = 0$, $W_V = 0$, $W_T = 1$. The node of the selected cluster which has the maximum rank gets the chance to become the CA for the transaction. Algorithm 10 shows the CA selection process.

Numerous applications provide arguments for categorizing the weighted priority in the various forms mentioned above. There are many real-time blockchain-based IoT applications like VANET [197] where delay (response time and validation time) plays a very crucial role compared to the trust factor we calculated from previous successful transactions. However, for financial applications trust is a more important issue [198] compared to delay. The three weighted factors can act like a tuning knob, depending on the application these weight values can be changed.

$$Rank = \begin{cases} W_R \times (1 - \frac{RT}{RT_{max}}), & \text{if, } W_V, W_T = 0 \\ W_V \times (1 - \frac{VT}{VT_{max}}), & \text{if, } W_R, W_T = 0 \\ W_T \times \frac{T}{T_{max}}, & \text{if, } W_R, W_V = 0 \\ \frac{1}{3} \times RT + \frac{1}{3} \times VT + \frac{1}{3} \times T, & \text{if,} W_R = W_V = W_T \end{cases} \tag{5.14}$$

Step-3

The smart contract verifies the selected node N as CA by N_{id}, and $N_{ETHaddress}$. If the verification process is successful then the node eligibility for becoming the CA is verified.

Step-4

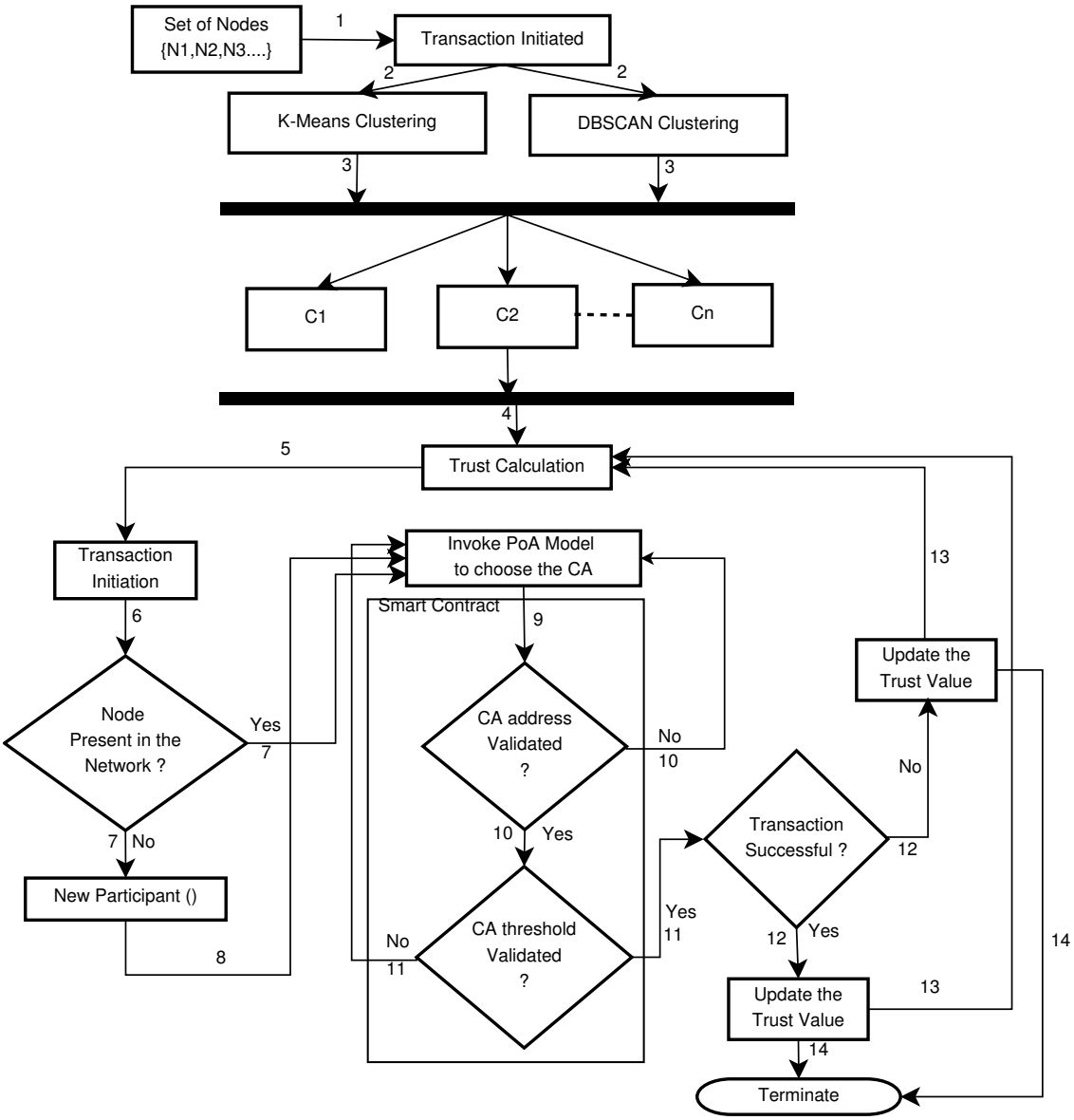

Figure 5.4: Workflow of the proposed CTB-PKI system

The selected expiry limit N_{Expiry} is compared with the revocation id N_{Rev}. If the N_{Rev} is found smaller than the N_{Rev} then only the CA is allowed to validate the transaction. Otherwise, the transaction initiator node is informed to select another CA.

Step-5

For every transaction, the CA_{Rev} is incremented by 1. In addition to the CA_{REV}, the trust value of the CA is reevaluated (see section 5.6.2.4).

Step-6

After a certain number of transactions, step 2 is invoked to reform the network cluster.

Algorithm 9 Proposed CTB-PKI

Input:Set of Nodes N=[N_1**,** N_2**,** N_3**, ...,** N_n**],** N_{id}**,** $N_{ETHaddress}$**,** N_{Rev}**,** N_{Expiry}
Output:Selected CA
Initiate Transaction
Invoke Proc $K-Means()$ and $DBSCAN()$
Define the optimal number of clusters k
Initiate m number of transactions with k clusters
Cluster selection process
Invoke $Selection()$ to select the appropriate CA
Invoke PoA()
for (i=1 to m) **do**
    get CA_{id}, $CA_{ETHaddress}$, CA_{Rev}, CA_{Expiry}
    Invoke Smart Contract to verify the identity of CA
    if ($CA_{id} == N_{id}$) **then**
        if ($CA_{ETHaddress} == N_{ETHaddress}$) **then**
            CA Identity verified
        else
            CA identity mismatched. Abort the transaction and select a new CA
        end if
    end if
    Invoke Smart Contract to check the eligibility of CA
    if ($CA_{Rev} \leq N_{Expiry}$) **then**
        Validate the Transaction
        $CA_{Rev}++$
    else
        Maximum Trial is over for the elected validator. Please select another node
    end if
    Calculate Trust of the CA
end for
Invoke $K-Means()$ for reclustering

5.7 Implementation and Performance Evaluation

The proposed clustering based PKIs are implemented in the open-source Ethereum platform ($GETH$). The *solidity* v 0.4.24 scripting language and $Truffle\ Suit$ are used to deploy the smart contract to the blockchain environment. A system with Windows 10 OS, 8GB RAM, Intel i5 with 2.8 GHz clock speed, 1TB HDD, and 500GB SSD is used to implement the proposed blockchain-based PKI.

5.7.1 Performance Evaluation of Approach 1

The blockchain network with 50 nodes is created with 300 transactions to evaluate the proposed clustering-based PKI. For determining the number of clusters using the K-Means algorithm with the elbow method as the internal validation method. Figure 5.5 shows the number of cluster selections using the K-Means. The figure clearly depicts that the SSE

Algorithm 10 CA Selection

Input:Set of Nodes N=[N_1, N_2, N_3, ..., N_n**],** N_{id}, $N_{ETHaddress}$, N_{Rev}, N_{Expiry}
Output:Selected CA
for (i=1 to k) **do**
  if ($RT_{avg(i)} < RT_{budget}$ & $VT_{avg(i)} < VT_{budget}$) **then**
    for (j=1 to N) **do**
      $Rank_j = W_R \times (1 - \frac{RT_i}{RT_{max}}) + W_V \times (1 - \frac{VT_i}{VT_{max}}) + W_T \times (1 - \frac{T_i}{T_{max}})$
    end for
  end if
end for

is forming an elbow with a continuous decrease in SSE value after the 3^{rd} cluster. So for the current research work, the number of preferred clusters is 3.

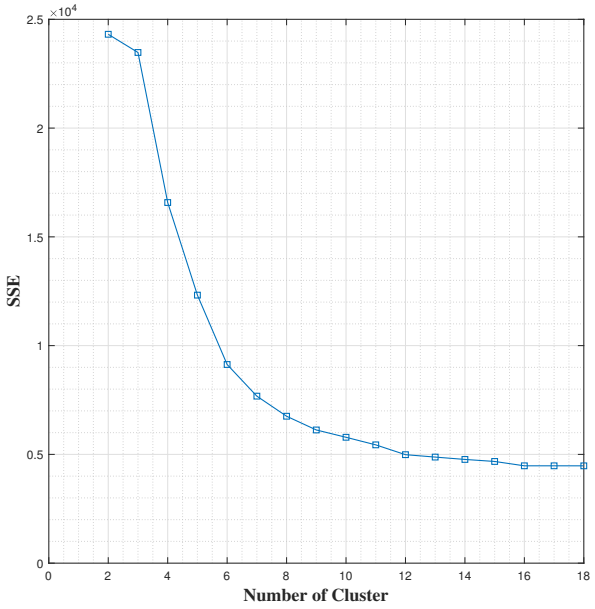

Figure 5.5: Workflow of the proposed CTB-PKI system

The performance of the blockchain in terms of RTF with clustering is compared with the performance of the network without the clustering technique. RTF i the total time required to complete the transaction starting from CA selection process to validation process. Figure 5.6 shows the performance comparison of a blockchain network with and without the clustering technique.

Figure 5.16 shows the gas utilization for the different transactions. The average gas utilization for the proposed PKI is 2.9×10^4.

The time-lapse of key generation and key validation is reported in figure 5.8 in contrast to the network scalability considering up to 50 number of nodes. This shows that upon increasing the size of the network, the key generation time and validation time also increase.

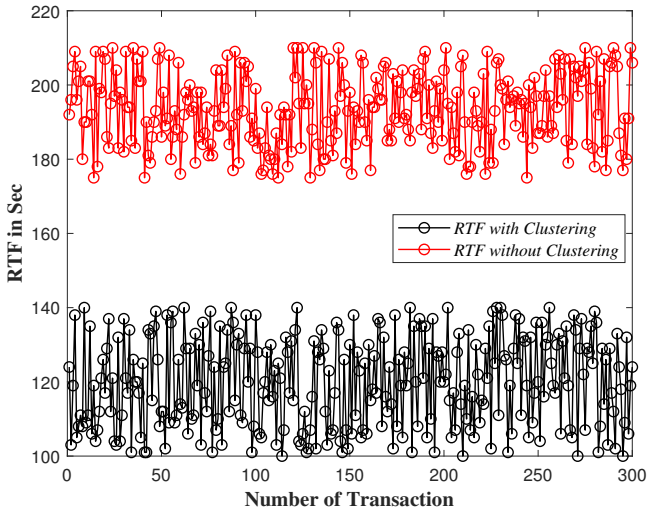

Figure 5.6: Workflow of the proposed CTB-PKI system

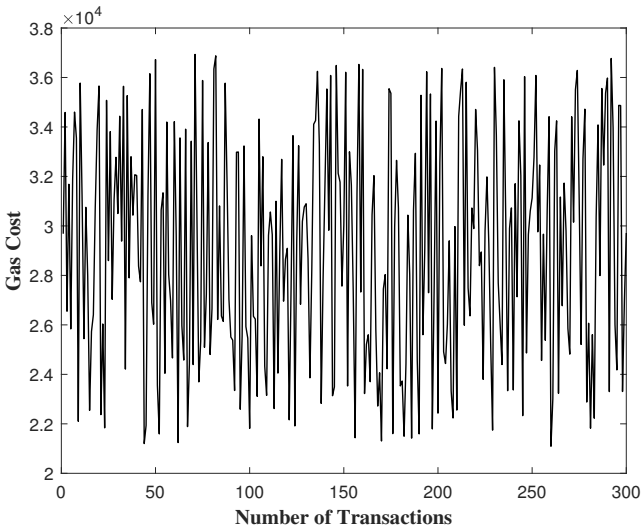

Figure 5.7: Workflow of the proposed CTB-PKI system

5.7.2 Performance Evaluation of Approach 2

The proposed $CTB - PKI$ is implemented in $GETH$ with 100 nodes. Each node is associated with $100ETH$ and a 4000000 Gas limit. In the Ethereum Ganache truffle suit the default gas limit for a node is 21000. However, different modules of the proposed PKI framework require more than 21000 gas. Therefore, the gas limit has been changed from the default value to the maximum limit. Lowering the gas limit causes a failure in the transaction.

17 iterations starting from 2 to 18 are performed for both the clustering algorithm to calculate the SC. The cluster that has the highest SC is considered the optimal number of clusters. From Figure 5.9, it is observed that the highest SC value for the number of clusters 2 is 0.69. So for the current work, the optimal number of clusters is taken as

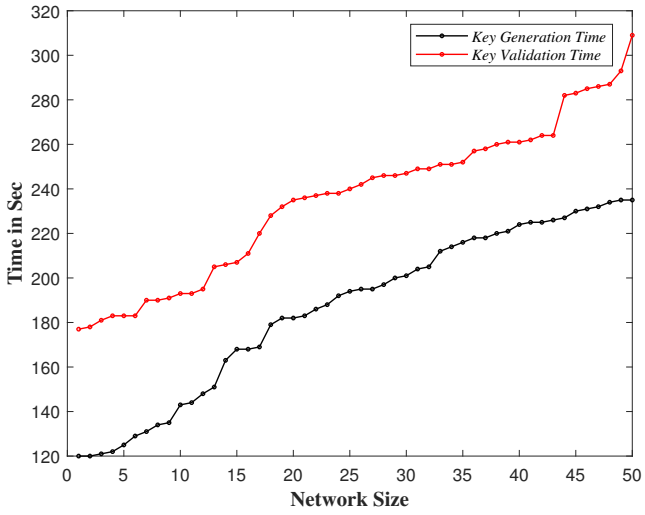

Figure 5.8: Workflow of the proposed CTB-PKI system

2 by using K-Means. Figure 5.10 shows the cluster formulation using the $K - Means$ algorithm with $\langle RT, VT \rangle$ as the input parameter.

```
For n_clusters = 2 The average silhouette_score is : 0.6903439735221752
For n_clusters = 3 The average silhouette_score is : 0.5993351086913034
For n_clusters = 4 The average silhouette_score is : 0.6779672730777585
For n_clusters = 5 The average silhouette_score is : 0.6232053822253856
For n_clusters = 6 The average silhouette_score is : 0.501762375116075
For n_clusters = 7 The average silhouette_score is : 0.4904752381240736
For n_clusters = 8 The average silhouette_score is : 0.4482692589275538
For n_clusters = 9 The average silhouette_score is : 0.42993449983178245
For n_clusters = 10 The average silhouette_score is : 0.45094886561184716
For n_clusters = 11 The average silhouette_score is : 0.3697487633279482
For n_clusters = 12 The average silhouette_score is : 0.36192718169148563
For n_clusters = 13 The average silhouette_score is : 0.37201351599685206
For n_clusters = 14 The average silhouette_score is : 0.3655032301819083
For n_clusters = 15 The average silhouette_score is : 0.3661721632155404
For n_clusters = 16 The average silhouette_score is : 0.36247638066038335
For n_clusters = 17 The average silhouette_score is : 0.3566815739540332
For n_clusters = 18 The average silhouette_score is : 0.3631375926968854
```

Figure 5.9: SH value for different clusters using K-Means

Similarly, 17 iterations starting from 2 to 18 are performed for both the clustering algorithm to calculate the SC using the DBSCAN algorithm. Figure 5.11 it can be observed that the highest SC value for cluster size 2 is ~ 0.46. So for the current work, the optimal number of clusters is taken as 2 by using DBSCAN as well. Figure 5.12 shows the cluster formulation using $DBSCAN$ algorithm with $\langle RT, VT \rangle$ as the input parameter.Table 5.5 shows the number of nodes present in each cluster with different clustering algorithms.

Figure 5.13 shows the number of clusters with RT, VT, and T as the input variable. Due to high computational time, the $DBSCAN$ algorithm is not used further. SC value for $k = 2$ is ~ 0.61 which is highest in contrast to other k value. The number of elements in clusters 0 and 1 is 61 and 39 respectively.

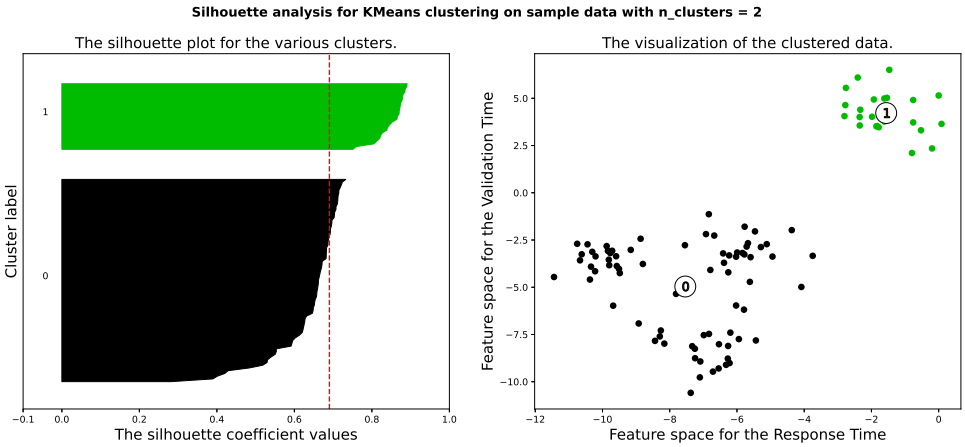

Figure 5.10: Number of Cluster using K-Means

```
For n_clusters = 2 The average silhouette_score is : 0.46578912980980918
For n_clusters = 3 The average silhouette_score is : 0.4414879065123001456
For n_clusters = 4 The average silhouette_score is : 0.418970972651880956
For n_clusters = 5 The average silhouette_score is : 0.409089872636177256
For n_clusters = 6 The average silhouette_score is : 0.4166728098726662
For n_clusters = 7 The average silhouette_score is : 0.408729119280092154
For n_clusters = 8 The average silhouette_score is : 0.3989305452237629567
For n_clusters = 9 The average silhouette_score is : 0.38565892556895
For n_clusters = 10 The average silhouette_score is : 0.39565681566519856
For n_clusters = 11 The average silhouette_score is : 0.37569812356006551
For n_clusters = 12 The average silhouette_score is : 0.365689151586515625
For n_clusters = 13 The average silhouette_score is : 0.32252616565622454
For n_clusters = 14 The average silhouette_score is : 0.35569815313698535
For n_clusters = 15 The average silhouette_score is : 0.32656165689516567
For n_clusters = 16 The average silhouette_score is : 0.31461811458485681
For n_clusters = 17 The average silhouette_score is : 0.30235693548528789
For n_clusters = 18 The average silhouette_score is : 0.295687886745098556
```

Figure 5.11: SH value for different clusters using K-Means

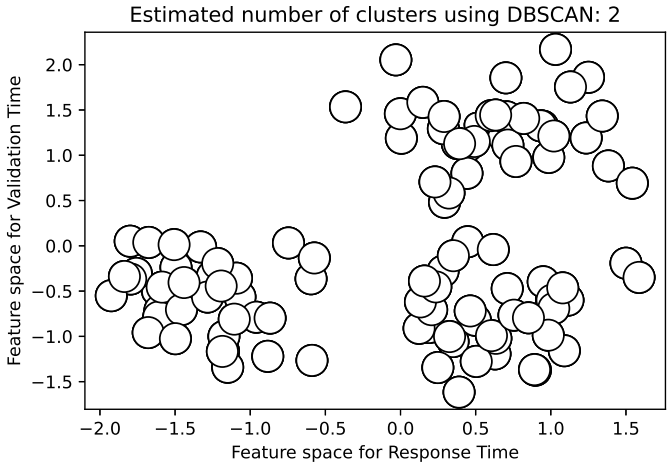

Figure 5.12: Number of clusters using DBSCAN algorithm

Figure 5.14 and Figure 5.15 reflect the response time and validation time of the proposed PKI with and without the clustering algorithm respectively. From the figure 5.15,

Table 5.5: Number of nodes in each cluster

Cluster	Algorithm		% of total node	
	K-Means	DBSCAN	K-Means	DBSCAN
Cluster 0	72	65	72 %	65%
Cluster 1	28	35	28%	35%
Total nodes	100	100	100%	100%

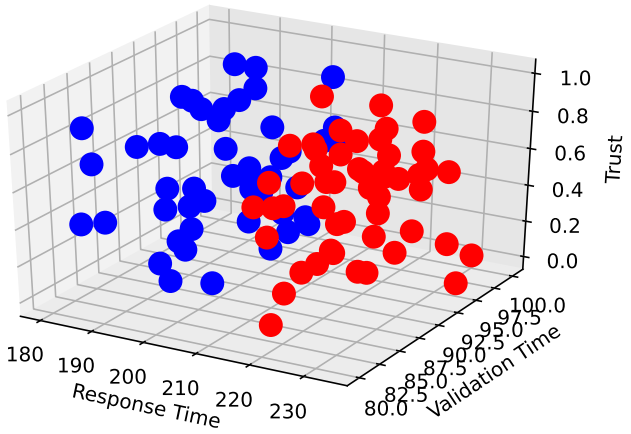

Figure 5.13: K-Means clustering with RT, VT, and Trust as the feature

it can be observed that the proposed work reduces RT about $\sim$ 38.2%. This improvement is due to the reduction of search space in the CA selection process. Figure 5.14 shows an improvement of VT with clustering in contrast to the VT without clustering. The proposed work reduces the VT about $\sim$ 2.2%. This improvement is because of the implication of trust value in selecting the CA for validating the transaction. Figure 5.16 shows the gas utilization with the different number of transactions of the proposed work. The average gas utilization of the proposed work is approximately 5×10^4.

The participant node needs to set its own budget by setting the corresponding weight factors W_R, W_V, and W_T. Depending upon the input the CA is selected with appropriate RT, VT, and T. Table 5.6 shows the CA selection ranking process for 10 transaction with different input budget. If the W_R is set to 1 then the node having the lowest RT value is considered as the CA. Accordingly, if the W_T is set to 1, then the node having the highest trust value within the cluster is selected as the CA.

5.7.3 Time Complexity Analysis

The proposed $CTB - PKI$ has different executable modules such as $New\ Participant$, $Validation$, $Signature\ Revocation$, $Smart\ Contract$,$K-Means$, and $DBSCAN$. Among

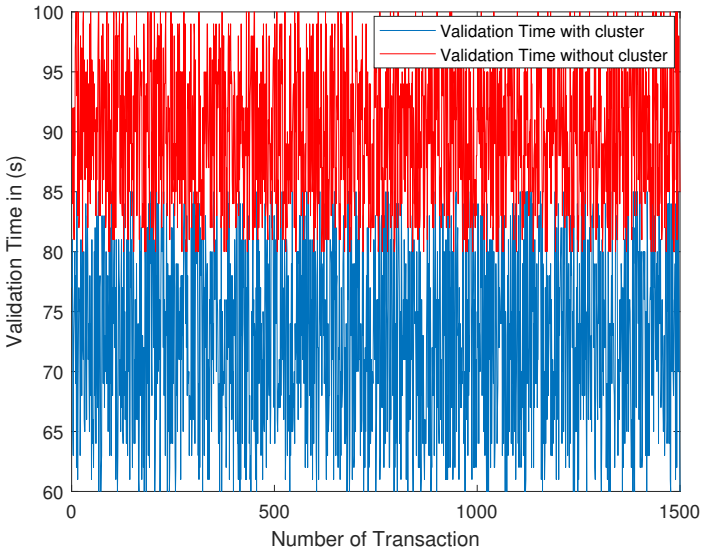

Figure 5.14: Validation time with and without cluster

Table 5.6: CA selection ranking based on the selected input budget

Transaction	Input Budget	RT (in Sec)	VT (in Sec)	T
T_1	$\langle 1,1,0 \rangle$	106	67	0.7
T_2	$\langle 1,0,0 \rangle$	111	71	0.47
T_3	$\langle 0,0,1 \rangle$	146	81	0.95
T_4	$\langle 1,0,1 \rangle$	112	79	0.85
T_5	$\langle 0,0,1 \rangle$	142	74	0.94
T_6	$\langle 1,1,0 \rangle$	108	69	0.37
T_7	$\langle 0,0,1 \rangle$	137	71	1
T_8	$\langle 1,1,1 \rangle$	112	69	1
T_9	$\langle 1,0,0 \rangle$	114	70	0.52
T_{10}	$\langle 1,1,1 \rangle$	119	75	0.81

these the *SmartContract* and *NewParticipant* modules have the time complexity of $O(n)$. Whereas the other two modules have constant time complexity $O(1)$. *NewParticipant* and *SmartContract* may receive multiple transactions thus making the time complexity of these two modules as $0(n)$. While for the other two modules *Validation* and the *SignatureRevocation* no transactional messages are generated, thus making the complexity of these two modules as $O(1)$. Implementing the PoA consensus mechanism has the time complexity $O(logn)$. Finally, the time complexity of $K-Means$ and $DBSCAN$ algorithm are $O(kN)$ and $O(N^2)$ with N as the number of nodes present in the network. The time complexity of each individual module is reported in Table 5.7.

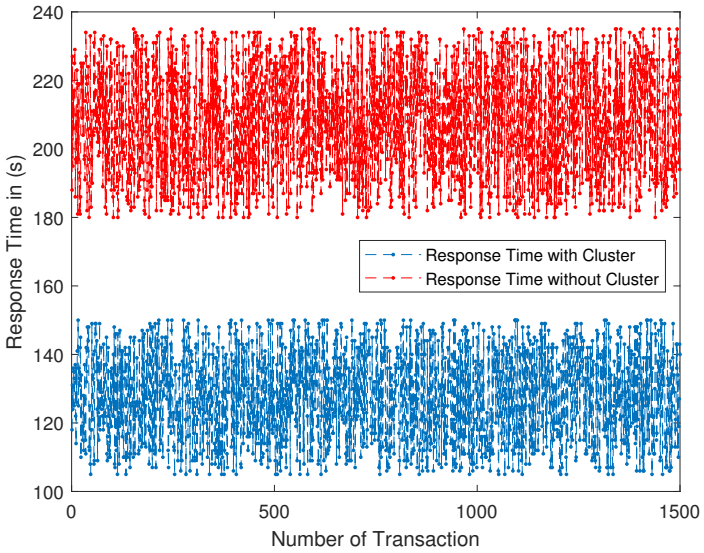

Figure 5.15: Response time with and without cluster

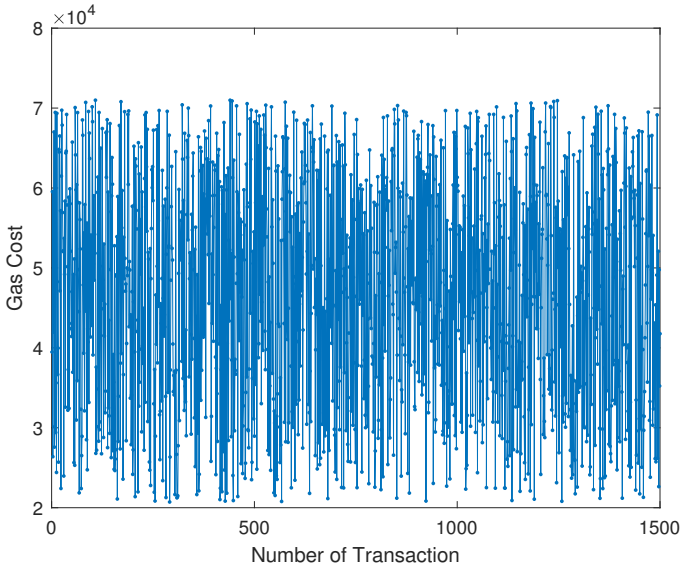

Figure 5.16: Gas utilization for different transactions

5.7.4 Critical Analysis

$BB-PKI$ in [186] proposed a PKI with the objective to avoid the $SPoF$ issue of the conventional PKI systems by introducing RAs. The node that wants a certificate for communication needs to forward the request to RA. RA then forwards the same request to the corresponding CA. With this solution, the proposed methods put a limitation on the P2P network concept. In $BlockPKI$ [187] a group of nodes is defined for becoming CA. For every transaction, the node that belongs to that group only can have the chance for becoming the CA which makes the whole process semi-decentralized.

In $Blockchain-basedPKImanagementframework$ [188] CA needs to store all the

Table 5.7: Time Complexity Analysis of proposed CTB-PKI model

Module	Time Complexity
New Participant	$O(n)$
Validation	$O(1)$
Signature Revocation	$O(1)$
Smart Contract	$O(n)$
PoA	$O(logn)$
K-Means	$O(kN)$
DBSCAN	$O(N^2)$

relevant information regarding the certificate issuance and revocation. This process needs high memory availability at the CA end which becomes the main issue of the proposed system. In *Cecoin* [189] the main issue is the adoption of PoW consensus mechanism for selecting the CA. PoW needs high computational capability at the node end which becomes the main issue for the lightweight clients in participating in the network communication. The limitation present behind the $X.509Cloud$ [190] is the number of certificates during each transaction as this PKI generates the different certificates for transaction and revocation. Table 5.8 shows the overall comparison of the above-mentioned blockchain-based PKIs in contrast to the proposed PKI system.

Table 5.8: Comparison of the proposed work with existing literature

PKI	Registration	Validation	Revocation	Trust Calculation	Node Clustering
[186]	✓	✓	✓	✗	✗
[187]	✓	✗	✓	✗	✗
[188]	✓	✓	✓	✗	✗
[189]	✓	✓	✓	✗	✗
[190]	✓	✓	✓	✗	✗
Proposed Work	✓	✓	✓	✓	✓

5.8 Conclusion

The proposed work addresses the limitation of the computational overhead of the existing PKI systems. This work reports a blockchain-based PKI system $CTB-PKI$ which uses clustering algorithms K-Means and DBSCAN to reduce the CA search space. The time complexity analysis shows that the K-Means algorithm is more suitable compared to the DBSCAN method for the current work. This work also focuses on the trust calculation of every participating node. The node, having a higher trust value and lower validation time,

and lower response time has a higher probability of becoming the CA for a transaction. For every successful transaction, the CA trustworthiness is increased and the trust value is decreased for every unsuccessful transaction. The performance of the proposed system is evaluated based on the response time, validation time, and gas utilization required for different transactions. The result analysis shows that network clustering puts an impact on response time and validation time. The proposed approach reduces $\sim$38.5% response time and $\sim$2.2% validation time compared to the PKI systems without clustering. The improvement in response time and validation time reduces transaction validation turnaround time in a blockchain-based communication system which makes the proposed $CTB-PKI$ more suitable for Blockchain 2.0 and 3.0 applications.

In our proposed $CTB-PKI$, the trust of every node is calculated based on successful and unsuccessful transactions. Other node communication quality factors such as data transmission rate and data delay rate can be considered for making the trust calculation more effective. The inclusion of clustering may increase the network performance by decreasing the latency such as RT and VT. However, the network energy consumption and computation effort have not been studied meticulously in our paper which we intend to address in our future studies. Moreover, we intend to improve the trust value of the proposed $CTB-PKI$ as well.

Chapter 6

Conclusions

The emergence of Blockchain-based dApps has addressed two fundamental issues present in centralized application systems: single-point failure and security risks using inherent features. The features of decentralization, immutability, and transparency in dApps enable them to effectively address the aforementioned issues. However, blockchain-based applications also face some challenging issues such as security concerns and computation overhead, which need to be effectively addressed.

The objective of the current thesis is to address the issues present in both centralized application systems and blockchain-based applications. The three primary contribution of the thesis is stated in Chapters 3, 4 and 5 respectively. Chapter 3 presents a basic blockchain-based decentralized application for storing and exchanging data of EHR among doctors and patients. This is a basic implementation to understand the working principle of blockchain-based applications. This dApp helps in removing the single-point failure of the centralized application system. However, dApp does not provide any effective way to deal with the cyber attacks present in the blockchain based application.

In Chapter 4, a security solution for blockchain-based PKI is developed. This BC-PKI prevents many popular threats like DoS, DDoS, MITM, 51%, Injection, Routing, and Eclipse attacks. Unlike conventional PKI, the developed PKI provides an effective way to identify malicious Certificate Authorities using its smart contract. In addition, this provides an equal opportunity to all available nodes to get Certificate Authority status. This is achieved by setting the threshold value of each node in order to become the Certificate Authority. If a node exceeds the predefined threshold, it will no longer be allowed to become a Certificate Authority. The DPoS consensus algorithm utilized in our PKI reduces timing complexity by avoiding excessive computational capacity at the nodes' end. The consensus mechanism and the adopted smart contract make the developed PKI affordable for lightweight applications. However, the developed PKI does not focus on reducing the computational overhead related to the Certificate Authority selection process.

To address this issue, Chapter 5 proposes a PKI that utilizes clustering approaches

based on validation time, response time, and trust. The developed PKI searches for Certificate Authorities on the nodes of the chosen cluster, rather than searching on all participant nodes, thereby reducing the search space for the Certificate Authority selection process. This work also focuses on the trust calculation of every participating node. The node, having a higher trust value, lower validation time, and lower response time, has a higher probability of becoming the Certificate Authority for a transaction. For every successful transaction, the trustworthiness of the Certificate Authority is increased, while the trust value is decreased for every unsuccessful transaction.

All the three works of this thesis is implemented in Ethereum blockchain environment $GETH$ which is association with *Ganache* $Truffle$ *Suit*. Author has noticed $GETH$ has a scalability issue in terms of the number of transactions and the number of nodes. An increase in the number of transactions and the node may affect the efficiency of the developed PKIs. However, the works as mentioned above have certain limitations stated as follows :

- The developed PKI of the current thesis calculates trust based on successful and unsuccessful transactions. Author feels considering communication quality factors of other nodes such as data transmission rate and data delay rate in trust calculation can make the proposed consensus algorithm more efficient.

- The implementations in this thesis work have tested upto 100 nodes because of the commercial constraints of Ethereum platform. In future, the author will study these implementations for large network traffic to identify its effectiveness.

- The network energy consumption and computation effort were not studied meticulously as they were beyond the scope of this thesis. However, this may be addressed in future works.

- All the designs of current thesis are implemented in Ethereum platform. The designs should be implemented in other blockchain platform as well to study the comparability, scalability and effectiveness.

In future, the author intends to develop a browser plug-in for implementing BC-PKI as mentioned in this thesis to identify the malicious Certificate Authority. Furthermore, the author will also study all the aforementioned limitations and provide feasible solutions for the same.

Chapter 7

Publications

7.1 Journals

[J.3] **A. Panigrahi**, A. K. Nayak, R. Paul, B. Sahu and S. Kant, "CTB-PKI: Clustering and Trust Enabled Blockchain Based PKI System for Efficient Communication in P2P Network," *in IEEE Access*, vol. 10, pp. 124277-124290, 2022, doi: 10.1109/ACCESS.2022.3222807.

[J.2] **A. Panigrahi**, A. K. Nayak, R. Paul, "Smart contract assisted blockchain based public key infrastructure system," *Transactions on Emerging Telecommunications Technologies*, Oct. 2022, doi: https://doi.org/10.1002/ett.4655.

[J.1] **A. Panigrahi**, A. K. Nayak, and R. Paul, "HealthCare EHR: A Blockchain-Based Decentralized Application." *International Journal of Information Systems and Supply Chain Management*, vol. 15, no. 3, pp. 1–15, Jul. 2022, doi: https://doi.org/10.4018/ijisscm.290017.

7.2 Conference

[C.2] **A. Panigrahi**, A. K. Nayak, R. Paul, "Impact of Clustering technique in enhancing the Blockchain network performance„" *IEEE*, 2022 International Conference on Machine Learning, Computer Systems and Security (MLCSS), Bhubaneswar, India, 2022, pp. 363-367, doi: 10.1109/MLCSS57186.2022.00072.

[C.1] **A. Panigrahi**, A. K. Nayak, R. Paul, "A Blockchain Based PKI System for Peer to Peer Network," *Lecture Notes in Networks and Systems*, 2nd International Conference on Advances in Distributed Computing and Machine Learning 2021, doi: https://doi.org/10.1002/ett.4655.

References

[1] N. Fedotova and L. Veltri, "Byzantine generals problem in the light of p2p computing," in *2006 3rd Annual International Conference on Mobile and Ubiquitous Systems-Workshops*, pp. 1–5, IEEE, 2006.

[2] H. Tenge and M. Okello, "Blockchain technology: Creating trust in a trustless environment," in *The Auditor's Guide to Blockchain Technology*, pp. 1–16, CRC Press.

[3] S. Nakamoto, "Bitcoin whitepaper," *URL: https://bitcoin. org/bitcoin. pdf-(: 17.07. 2019)*, 2008.

[4] A. Deshpande, K. Stewart, L. Lepetit, and S. Gunashekar, "Distributed ledger technologies/blockchain: Challenges, opportunities and the prospects for standards," *Overview report The British Standards Institution (BSI)*, vol. 40, p. 40, 2017.

[5] J. Abou Jaoude and R. G. Saade, "Blockchain applications–usage in different domains," *Ieee Access*, vol. 7, pp. 45360–45381, 2019.

[6] Y. Guo and C. Liang, "Blockchain application and outlook in the banking industry," *Financial innovation*, vol. 2, pp. 1–12, 2016.

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

[8] M. Dabbagh, M. Sookhak, and N. S. Safa, "The evolution of blockchain: A bibliometric study," *IEEE access*, vol. 7, pp. 19212–19221, 2019.

[9] D. Guegan, "Public blockchain versus private blockhain," 2017.

[10] J. Weise, "Public key infrastructure overview," *Sun BluePrints OnLine, August*, pp. 1–27, 2001.

[11] S. Kiran, P. Lareau, and S. Lloyd, "Pki basics-a technical perspective," in *PKI-Forum (http://www. pkiforum. org)*, 2002.

[12] Y. Li, "Emerging blockchain-based applications and techniques," *Service Oriented Computing and Applications*, vol. 13, pp. 279–285, 2019.

[13] X. Xu, Q. Lu, Y. Liu, L. Zhu, H. Yao, and A. V. Vasilakos, "Designing blockchain-based applications a case study for imported product traceability," *Future Generation Computer Systems*, vol. 92, pp. 399–406, 2019.

[14] P. Sharma, R. Jindal, and M. D. Borah, "A review of blockchain-based applications and challenges," *Wireless Personal Communications*, pp. 1–43, 2022.

[15] T. Hewa, M. Ylianttila, and M. Liyanage, "Survey on blockchain based smart contracts: Applications, opportunities and challenges," *Journal of Network and Computer Applications*, vol. 177, p. 102857, 2021.

[16] I. A. Omar, H. R. Hasan, R. Jayaraman, K. Salah, and M. Omar, "Implementing decentralized auctions using blockchain smart contracts," *Technological Forecasting and Social Change*, vol. 168, p. 120786, 2021.

[17] Y. Lu, P. Li, and H. Xu, "A food anti-counterfeiting traceability system based on blockchain and internet of things," *Procedia Computer Science*, vol. 199, pp. 629–636, 2022.

[18] R. Kumar, P. Kumar, R. Tripathi, G. P. Gupta, S. Garg, and M. M. Hassan, "A distributed intrusion detection system to detect ddos attacks in blockchain-enabled iot network," *Journal of Parallel and Distributed Computing*, vol. 164, pp. 55–68, 2022.

[19] J. Yin, Y. Xiao, Q. Pei, Y. Ju, L. Liu, M. Xiao, and C. Wu, "Smartdid: a novel privacy-preserving identity based on blockchain for iot," *IEEE Internet of Things Journal*, 2022.

[20] N. A. Ugochukwu, S. Goyal, and S. Arumugam, "Blockchain-based iot-enabled system for secure and efficient logistics management in the era of ir 4.0," *Journal of Nanomaterials*, vol. 2022, 2022.

[21] W. Powell, M. Foth, S. Cao, and V. Natanelov, "Garbage in garbage out: The precarious link between iot and blockchain in food supply chains," *Journal of Industrial Information Integration*, vol. 25, p. 100261, 2022.

[22] E. Bandara, D. Tosh, P. Foytik, S. Shetty, N. Ranasinghe, and K. De Zoysa, "Tikiri—towards a lightweight blockchain for iot," *Future Generation Computer Systems*, vol. 119, pp. 154–165, 2021.

[23] N. Andola, S. Venkatesan, S. Verma, *et al.*, "Poewal: A lightweight consensus mechanism for blockchain in iot," *Pervasive and Mobile Computing*, vol. 69, p. 101291, 2020.

[24] M. A. Bouras, Q. Lu, S. Dhelim, and H. Ning, "A lightweight blockchain-based iot identity management approach," *Future Internet*, vol. 13, no. 2, p. 24, 2021.

[25] J.-H. Lee, "Bidaas: Blockchain based id as a service," *IEEE Access*, vol. 6, pp. 2274–2278, 2017.

[26] J.-H. Hsiao, R. Tso, C.-M. Chen, and M.-E. Wu, "Decentralized e-voting systems based on the blockchain technology," in *Advances in Computer Science and Ubiquitous Computing: CSA-CUTE 17*, pp. 305–309, Springer, 2018.

[27] C. Sullivan and E. Burger, "E-residency and blockchain," *computer law & security review*, vol. 33, no. 4, pp. 470–481, 2017.

[28] T. Moura and A. Gomes, "Blockchain voting and its effects on election transparency and voter confidence," in *Proceedings of the 18th annual international conference on digital government research*, pp. 574–575, 2017.

[29] F. Þ. Hjálmarsson, G. K. Hreiðarsson, M. Hamdaqa, and G. Hjálmtỳsson, "Blockchain-based e-voting system," in *2018 IEEE 11th international conference on cloud computing (CLOUD)*, pp. 983–986, IEEE, 2018.

[30] H. Hou, "The application of blockchain technology in e-government in china," in *2017 26th International Conference on Computer Communication and Networks (ICCCN)*, pp. 1–4, IEEE, 2017.

[31] S. N. Khan, M. Shael, and M. Majdalawieh, "Blockchain technology as a support infrastructure in e-government evolution at dubai economic department," in *Proceedings of the 1st International Electronics Communication Conference*, pp. 124–130, 2019.

[32] R. Páez, M. Pérez, G. Ramírez, J. Montes, and L. Bouvarel, "An architecture for biometric electronic identification document system based on blockchain," *Future Internet*, vol. 12, no. 1, p. 10, 2020.

[33] L. Liu, C. Piao, X. Jiang, and L. Zheng, "Research on governmental data sharing based on local differential privacy approach," in *2018 IEEE 15th international conference on e-business engineering (ICEBE)*, pp. 39–45, IEEE, 2018.

[34] M. E. Ghanem and A. Alsoufi, "Interoperable framework to enhance citizen services in the kingdom of bahrain," in *2019 International Conference on Innovation and Intelligence for Informatics, Computing, and Technologies (3ICT)*, pp. 1–4, IEEE, 2019.

[35] Y. Zhang, S. Deng, Y. Zhang, and J. Kong, "Research on government information sharing model using blockchain technology," in *2019 10th International Conference on Information Technology in Medicine and Education (ITME)*, pp. 726–729, IEEE, 2019.

[36] N.-H. Nguyen, B. M. Nguyen, T.-C. Dao, and B.-L. Do, "Towards blockchainizing land valuation certificate management procedures in vietnam," in *2020 RIVF International Conference on Computing and Communication Technologies (RIVF)*, pp. 1–6, IEEE, 2020.

[37] H. Huang, X. Sun, F. Xiao, P. Zhu, and W. Wang, "Blockchain-based ehealth system for auditable ehrs manipulation in cloud environments," *Journal of Parallel and Distributed Computing*, vol. 148, pp. 46–57, 2021.

[38] Y. Zhuang, L. R. Sheets, Y.-W. Chen, Z.-Y. Shae, J. J. Tsai, and C.-R. Shyu, "A patient-centric health information exchange framework using blockchain technology," *IEEE journal of biomedical and health informatics*, vol. 24, no. 8, pp. 2169–2176, 2020.

[39] H. Guo, W. Li, E. Meamari, C.-C. Shen, and M. Nejad, "Attribute-based multisignature and encryption for ehr management: A blockchain-based solution," in *2020 IEEE International Conference on Blockchain and Cryptocurrency (ICBC)*, pp. 1–5, IEEE, 2020.

[40] M. Kim, S. Yu, J. Lee, Y. Park, and Y. Park, "Design of secure protocol for cloud-assisted electronic health record system using blockchain," *Sensors*, vol. 20, no. 10, p. 2913, 2020.

[41] S. Chenthara, K. Ahmed, H. Wang, F. Whittaker, and Z. Chen, "Healthchain: A novel framework on privacy preservation of electronic health records using blockchain technology," *Plos one*, vol. 15, no. 12, p. e0243043, 2020.

[42] L. Chen, W.-K. Lee, C.-C. Chang, K.-K. R. Choo, and N. Zhang, "Blockchain based searchable encryption for electronic health record sharing," *Future generation computer systems*, vol. 95, pp. 420–429, 2019.

[43] A. Karmakar, P. Ghosh, P. S. Banerjee, and D. De, "Chainsure: Agent free insurance system using blockchain for healthcare 4.0," *Intelligent Systems with Applications*, p. 200177, 2023.

[44] G. Saldamli, V. Reddy, K. S. Bojja, M. K. Gururaja, Y. Doddaveerappa, and L. Tawalbeh, "Health care insurance fraud detection using blockchain," in *2020*

Seventh international conference on software defined systems (SDS), pp. 145–152, IEEE, 2020.

[45] T. K. Mackey, K. Miyachi, D. Fung, S. Qian, and J. Short, "Combating health care fraud and abuse: Conceptualization and prototyping study of a blockchain antifraud framework," *Journal of medical Internet research*, vol. 22, no. 9, p. e18623, 2020.

[46] T.-T. Kuo, H.-E. Kim, and L. Ohno-Machado, "Blockchain distributed ledger technologies for biomedical and health care applications," *Journal of the American Medical Informatics Association*, vol. 24, no. 6, pp. 1211–1220, 2017.

[47] A. S. Rajawat, P. Bedi, S. Goyal, R. N. Shaw, A. Ghosh, and S. Aggarwal, "Ai and blockchain for healthcare data security in smart cities," *AI and IoT for Smart City Applications*, pp. 185–198, 2022.

[48] C. Schinckus, "A nuanced perspective on blockchain technology and healthcare," *Technology in Society*, vol. 71, p. 102082, 2022.

[49] A. P. Singh, N. R. Pradhan, A. K. Luhach, S. Agnihotri, N. Z. Jhanjhi, S. Verma, U. Ghosh, D. S. Roy, *et al.*, "A novel patient-centric architectural framework for blockchain-enabled healthcare applications," *IEEE Transactions on Industrial Informatics*, vol. 17, no. 8, pp. 5779–5789, 2020.

[50] J. Angelis and E. R. Da Silva, "Blockchain adoption: A value driver perspective," *Business Horizons*, vol. 62, no. 3, pp. 307–314, 2019.

[51] S. Angraal, H. M. Krumholz, and W. L. Schulz, "Blockchain technology: applications in health care," *Circulation: Cardiovascular quality and outcomes*, vol. 10, no. 9, p. e003800, 2017.

[52] G. G. Dagher, J. Mohler, M. Milojkovic, and P. B. Marella, "Ancile: Privacy-preserving framework for access control and interoperability of electronic health records using blockchain technology," *Sustainable cities and society*, vol. 39, pp. 283–297, 2018.

[53] D. Dhagarra, M. Goswami, P. Sarma, and A. Choudhury, "Big data and blockchain supported conceptual model for enhanced healthcare coverage: The indian context," *Business Process Management Journal*, 2019.

[54] H. Hu, J. Xu, M. Liu, and M. K. Lim, "Vaccine supply chain management: An intelligent system utilizing blockchain, iot and machine learning," *Journal of Business Research*, vol. 156, p. 113480, 2023.

[55] D. Agrawal, S. Minocha, S. Namasudra, and A. H. Gandomi, "A robust drug recall supply chain management system using hyperledger blockchain ecosystem," *Computers in biology and medicine*, vol. 140, p. 105100, 2022.

[56] I. Ehsan, M. Irfan Khalid, L. Ricci, J. Iqbal, A. Alabrah, S. Sajid Ullah, and T. M. Alfakih, "A conceptual model for blockchain-based agriculture food supply chain system," *Scientific Programming*, vol. 2022, pp. 1–15, 2022.

[57] S. A. Bhat, N.-F. Huang, I. B. Sofi, and M. Sultan, "Agriculture-food supply chain management based on blockchain and iot: A narrative on enterprise blockchain interoperability," *Agriculture*, vol. 12, no. 1, p. 40, 2021.

[58] M. P. Caro, M. S. Ali, M. Vecchio, and R. Giaffreda, "Blockchain-based traceability in agri-food supply chain management: A practical implementation," in *2018 IoT Vertical and Topical Summit on Agriculture-Tuscany (IOT Tuscany)*, pp. 1–4, IEEE, 2018.

[59] A. Shahid, A. Almogren, N. Javaid, F. A. Al-Zahrani, M. Zuair, and M. Alam, "Blockchain-based agri-food supply chain: A complete solution," *Ieee Access*, vol. 8, pp. 69230–69243, 2020.

[60] S. Ahmed and N. t. Broek, "Blockchain could boost food security," *Nature*, vol. 550, no. 7674, pp. 43–43, 2017.

[61] R. Singh, A. D. Dwivedi, G. Srivastava, P. Chatterjee, and J. C.-W. Lin, "A privacy preserving internet of things smart healthcare financial system," *IEEE Internet of Things Journal*, 2023.

[62] B. Son and H. Jang, "Economics of blockchain-based securities settlement," *Research in International Business and Finance*, vol. 64, p. 101842, 2023.

[63] J. Saxena, "Block chain architecture in financial system for integrity, transparency, and trust-free transaction," in *Blockchain for IoT*, pp. 109–126, Chapman and Hall/CRC, 2023.

[64] R. R. Chen, K. Chen, and C. X. Ou, "Facilitating interorganizational trust in strategic alliances by leveraging blockchain-based systems: Case studies of two eastern banks," *International Journal of Information Management*, vol. 68, p. 102521, 2023.

[65] J. C. Chuah, "Money laundering considerations in blockchain-based maritime trade and commerce," *European Journal of Risk Regulation*, vol. 14, no. 1, pp. 49–64, 2023.

[66] R. Jiang, Y. Kang, Y. Liu, Z. Liang, Y. Duan, Y. Sun, and J. Liu, "A trust transitivity model of small and medium-sized manufacturing enterprises under blockchain-based supply chain finance," *International Journal of Production Economics*, vol. 247, p. 108469, 2022.

[67] Q. Zhang and T. Liao, "The construction of college intelligent financial system based on blockchain technology," in *2022 International Conference on Artificial Intelligence and Autonomous Robot Systems (AIARS)*, pp. 139–142, IEEE, 2022.

[68] T. Zhang and Z. Huang, "Blockchain and central bank digital currency," *ICT Express*, vol. 8, no. 2, pp. 264–270, 2022.

[69] P. Patil, M. Sangeetha, and V. Bhaskar, "A consortium blockchain based overseas fund transfer system," *Wireless Personal Communications*, vol. 122, no. 2, pp. 1367–1389, 2022.

[70] P. Ranjan, B. Sharma, A. Mittal, H. Gupta, and A. K. Singh, "Blockchain powered government financial system," in *2022 International Conference for Advancement in Technology (ICONAT)*, pp. 1–6, IEEE, 2022.

[71] J. Fu, B. Cao, X. Wang, P. Zeng, W. Liang, and Y. Liu, "Bfs: A blockchain-based financing scheme for logistics company in supply chain finance," *Connection Science*, vol. 34, no. 1, pp. 1929–1955, 2022.

[72] N. Kabra, P. Bhattacharya, S. Tanwar, and S. Tyagi, "Mudrachain: Blockchain-based framework for automated cheque clearance in financial institutions," *Future Generation Computer Systems*, vol. 102, pp. 574–587, 2020.

[73] S. B. Patel, P. Bhattacharya, S. Tanwar, and N. Kumar, "Kirti: A blockchain-based credit recommender system for financial institutions," *IEEE Transactions on Network Science and Engineering*, vol. 8, no. 2, pp. 1044–1054, 2020.

[74] F. Kausar, F. M. Senan, H. M. Asif, and K. Raahemifar, "6g technology and taxonomy of attacks on blockchain technology," *Alexandria Engineering Journal*, vol. 61, no. 6, pp. 4295–4306, 2022.

[75] R. Chaganti, B. Bhushan, and V. Ravi, "A survey on blockchain solutions in ddos attacks mitigation: Techniques, open challenges and future directions," *Computer Communications*, 2022.

[76] M. Saad, A. Khormali, and A. Mohaisen, "End-to-end analysis of in-browser cryptojacking," *arXiv preprint arXiv:1809.02152*, 2018.

[77] F. A. Aponte-Novoa, A. L. S. Orozco, R. Villanueva-Polanco, and P. Wightman, "The 51% attack on blockchains: A mining behavior study," *IEEE Access*, vol. 9, pp. 140549–140564, 2021.

[78] J. Sengupta, S. Ruj, and S. D. Bit, "A comprehensive survey on attacks, security issues and blockchain solutions for iot and iiot," *Journal of Network and Computer Applications*, vol. 149, p. 102481, 2020.

[79] B. Alangot, D. Reijsbergen, S. Venugopalan, P. Szalachowski, and K. S. Yeo, "Decentralized and lightweight approach to detect eclipse attacks on proof of work blockchains," *IEEE Transactions on Network and Service Management*, vol. 18, no. 2, pp. 1659–1672, 2021.

[80] X. Fu, H. Wang, and P. Shi, "A survey of blockchain consensus algorithms: mechanism, design and applications," *Science China Information Sciences*, vol. 64, pp. 1–15, 2021.

[81] S. Aggarwal and N. Kumar, "Attacks on blockchain," in *Advances in Computers*, vol. 121, pp. 399–410, Elsevier, 2021.

[82] G. Morganti, E. Schiavone, and A. Bondavalli, "Risk assessment of blockchain technology," in *2018 Eighth Latin-American Symposium on Dependable Computing (LADC)*, pp. 87–96, IEEE, 2018.

[83] A. Averin and O. Averina, "Review of blockchain technology vulnerabilities and blockchain-system attacks," in *2019 International Multi-Conference on Industrial Engineering and Modern Technologies (FarEastCon)*, pp. 1–6, IEEE, 2019.

[84] S. Bouraga, "A taxonomy of blockchain consensus protocols: A survey and classification framework," *Expert Systems with Applications*, vol. 168, p. 114384, 2021.

[85] Y. Kwon, D. Kim, Y. Son, E. Vasserman, and Y. Kim, "Be selfish and avoid dilemmas: Fork after withholding (faw) attacks on bitcoin," in *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*, pp. 195–209, 2017.

[86] P. Wei, Q. Yuan, and Y. Zheng, "Security of the blockchain against long delay attack," in *Advances in Cryptology–ASIACRYPT 2018: 24th International Conference on the Theory and Application of Cryptology and Information Security, Brisbane, QLD, Australia, December 2–6, 2018, Proceedings, Part III 24*, pp. 250–275, Springer, 2018.

[87] A. Garba, D. Khoury, P. Balian, S. Haddad, J. Sayah, Z. Chen, Z. Guan, H. Hamdan, J. Charafeddine, and K. Al-Mutib, "Lightcert4iots: Blockchain-based lightweight certificates authentication for iot applications," *IEEE Access*, vol. 11, pp. 28370–28383, 2023.

[88] Z. Zhai, S. Shen, and Y. Mao, "Bpki: A secure and scalable blockchain-based public key infrastructure system for web services," *Journal of Information Security and Applications*, vol. 68, p. 103226, 2022.

[89] S. Wang, H. Li, J. Chen, J. Wang, and Y. Deng, "Dag blockchain-based lightweight authentication and authorization scheme for iot devices," *Journal of Information Security and Applications*, vol. 66, p. 103134, 2022.

[90] X. Luo, Z. Xu, K. Xue, Q. Jiang, R. Li, and D. Wei, "Scalacert: Scalability-oriented pki with redactable consortium blockchain enabled" on-cert" certificate revocation," in *2022 IEEE 42nd International Conference on Distributed Computing Systems (ICDCS)*, pp. 1236–1246, IEEE, 2022.

[91] C.-G. Koa, S.-H. Heng, and J.-J. Chin, "Etherst: Ethereum-based public key infrastructure identity management with a reward-and-punishment mechanism," *Symmetry*, vol. 13, no. 9, p. 1640, 2021.

[92] J. Xie, X. Tan, and L. Tan, "Cr-ba: Public key infrastructure certificate revocation scheme based on blockchain and accumulator," *Security and Communication Networks*, vol. 2022, 2022.

[93] X. Ge, L. Wang, W. An, X. Zhou, and B. Li, "Crchain: An efficient certificate revocation scheme based on blockchain," in *Algorithms and Architectures for Parallel Processing: 21st International Conference, ICA3PP 2021, Virtual Event, December 3–5, 2021, Proceedings, Part II*, pp. 453–472, Springer, 2022.

[94] D. Moussaoui, B. Kadri, M. Feham, and B. A. Bensaber, "A distributed blockchain based pki (bcpki) architecture to enhance privacy in vanet," in *2020 2nd International Workshop on Human-Centric Smart Environments for Health and Well-being (IHSH)*, pp. 75–79, IEEE, 2021.

[95] I. A. Obiri, J. Yang, Q. Xia, and J. Gao, "A sovereign pki for iot devices based on the blockchain technology," in *2021 18th International Computer Conference on Wavelet Active Media Technology and Information Processing (ICCWAMTIP)*, pp. 110–115, IEEE, 2021.

[96] C. Zhang, L. Zhu, and C. Xu, "Bpaf: Blockchain-enabled reliable and privacy-preserving authentication for fog-based iot devices," *IEEE Consumer Electronics Magazine*, vol. 11, no. 2, pp. 88–96, 2021.

[97] A. Garba, Z. Chen, Z. Guan, and G. Srivastava, "Lightledger: a novel blockchain-based domain certificate authentication and validation scheme," *IEEE Transactions on Network Science and Engineering*, vol. 8, no. 2, pp. 1698–1710, 2021.

[98] T. Sermpinis, G. Vlahavas, K. Karasavvas, and A. Vakali, "Detract: a decentralized, transparent, immutable and open pki certificate framework," *International Journal of Information Security*, vol. 20, pp. 553–570, 2021.

[99] R. G. Shukla, A. Agarwal, and S. Shukla, "Blockchain-powered smart healthcare system," in *Handbook of research on blockchain technology*, pp. 245–270, Elsevier, 2020.

[100] A. A. Abdellatif, A. Z. Al-Marridi, A. Mohamed, A. Erbad, C. F. Chiasserini, and A. Refaey, "sshealth: toward secure, blockchain-enabled healthcare systems," *IEEE Network*, vol. 34, no. 4, pp. 312–319, 2020.

[101] F. A. Khan, M. Asif, A. Ahmad, M. Alharbi, and H. Aljuaid, "Blockchain technology, improvement suggestions, security challenges on smart grid and its application in healthcare for sustainable development," *Sustainable Cities and Society*, vol. 55, p. 102018, 2020.

[102] S. Tanwar, K. Parekh, and R. Evans, "Blockchain-based electronic healthcare record system for healthcare 4.0 applications," *Journal of Information Security and Applications*, vol. 50, p. 102407, 2020.

[103] P. Zhang, J. White, D. C. Schmidt, and G. Lenz, "Design of blockchain-based apps using familiar software patterns to address interoperability challenges in healthcare," in *PLoP-24th Conference On Pattern Languages Of Programs*, 2017.

[104] P. Zhang, J. White, D. C. Schmidt, G. Lenz, and S. T. Rosenbloom, "Fhirchain: applying blockchain to securely and scalably share clinical data," *Computational and structural biotechnology journal*, vol. 16, pp. 267–278, 2018.

[105] F. Blum, B. Severin, M. Hettmer, P. Hückinghaus, and V. Gruhn, "Building hybrid dapps using blockchain tactics-the meta-transaction example," in *2020 IEEE International Conference on Blockchain and Cryptocurrency (ICBC)*, pp. 1–5, IEEE, 2020.

[106] W. Cai, Z. Wang, J. B. Ernst, Z. Hong, C. Feng, and V. C. Leung, "Decentralized applications: The blockchain-empowered software system," *IEEE access*, vol. 6, pp. 53019–53033, 2018.

[107] 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, pp. 62–75, 2019.

[108] P. Ratta, A. Kaur, S. Sharma, M. Shabaz, and G. Dhiman, "Application of blockchain and internet of things in healthcare and medical sector: applications, challenges, and future perspectives," *Journal of Food Quality*, vol. 2021, pp. 1–20, 2021.

[109] A. A. Siyal, A. Z. Junejo, M. Zawish, K. Ahmed, A. Khalil, and G. Soursou, "Applications of blockchain technology in medicine and healthcare: Challenges and future perspectives," *Cryptography*, vol. 3, no. 1, p. 3, 2019.

[110] A. Al Omar, M. Z. A. Bhuiyan, A. Basu, S. Kiyomoto, and M. S. Rahman, "Privacy-friendly platform for healthcare data in cloud based on blockchain environment," *Future generation computer systems*, vol. 95, pp. 511–521, 2019.

[111] C. C. Agbo, Q. H. Mahmoud, and J. M. Eklund, "Blockchain technology in healthcare: a systematic review," in *Healthcare*, vol. 7, p. 56, MDPI, 2019.

[112] K. P. Satamraju, "Proof of concept of scalable integration of internet of things and blockchain in healthcare," *Sensors*, vol. 20, no. 5, p. 1389, 2020.

[113] M. Zarour, M. T. J. Ansari, M. Alenezi, A. K. Sarkar, M. Faizan, A. Agrawal, R. Kumar, and R. A. Khan, "Evaluating the impact of blockchain models for secure and trustworthy electronic healthcare records," *IEEE Access*, vol. 8, pp. 157959–157973, 2020.

[114] M. P. Singh and A. K. Chopra, "Computational governance and violable contracts for blockchain applications," *Computer*, vol. 53, no. 1, pp. 53–62, 2020.

[115] S. Wang, J. Wang, X. Wang, T. Qiu, Y. Yuan, L. Ouyang, Y. Guo, and F.-Y. Wang, "Blockchain-powered parallel healthcare systems based on the acp approach," *IEEE Transactions on Computational Social Systems*, vol. 5, no. 4, pp. 942–950, 2018.

[116] Y. Sun, R. Zhang, X. Wang, K. Gao, and L. Liu, "A decentralizing attribute-based signature for healthcare blockchain," in *2018 27th International conference on computer communication and networks (ICCCN)*, pp. 1–9, IEEE, 2018.

[117] S. Bryatov and A. Borodinov, "Blockchain technology in the pharmaceutical supply chain: Researching a business model based on hyperledger fabric," in *Proceedings of the International Conference on Information Technology and Nanotechnology (ITNT), Samara, Russia*, vol. 10, pp. 1613–0073, 2019.

[118] A. Khatoon, "A blockchain-based smart contract system for healthcare management," *Electronics*, vol. 9, no. 1, p. 94, 2020.

[119] S. Figueroa, J. Añorga, and S. Arrizabalaga, "An attribute-based access control model in rfid systems based on blockchain decentralized applications for healthcare environments," *Computers*, vol. 8, no. 3, p. 57, 2019.

[120] Y. Zhuang, Y.-W. Chen, Z.-Y. Shae, and C.-R. Shyu, "Generalizable layered blockchain architecture for health care applications: development, case studies, and evaluation," *Journal of Medical Internet Research*, vol. 22, no. 7, p. e19029, 2020.

[121] K. N. Griggs, O. Ossipova, C. P. Kohlios, A. N. Baccarini, E. A. Howson, and T. Hayajneh, "Healthcare blockchain system using smart contracts for secure automated remote patient monitoring," *Journal of medical systems*, vol. 42, pp. 1–7, 2018.

[122] T. Justinia, "Blockchain technologies: opportunities for solving real-world problems in healthcare and biomedical sciences," *Acta Informatica Medica*, vol. 27, no. 4, p. 284, 2019.

[123] S. S. Kavathekar and R. Patil, "Data sharing and privacy–preserving of medical records using blockchain," in *Sustainable Communication Networks and Application: ICSCN 2019*, pp. 65–72, Springer, 2020.

[124] A. Azaria, A. Ekblaw, T. Vieira, and A. Lippman, "Medrec: Using blockchain for medical data access and permission management," in *2016 2nd international conference on open and big data (OBD)*, pp. 25–30, IEEE, 2016.

[125] L. Zhou, L. Wang, and Y. Sun, "Mistore: a blockchain-based medical insurance storage system," *Journal of medical systems*, vol. 42, no. 8, p. 149, 2018.

[126]

[127] D. V. Dimitrov, "Blockchain applications for healthcare data management," *Healthcare informatics research*, vol. 25, no. 1, pp. 51–56, 2019.

[128] 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, pp. 841–853, 2020.

[129] Y. Wen, F. Lu, Y. Liu, and X. Huang, "Attacks and countermeasures on blockchains: A survey from layering perspective," *Computer Networks*, vol. 191, p. 107978, 2021.

[130] S. Singh, A. S. Hosen, and B. Yoon, "Blockchain security attacks, challenges, and solutions for the future distributed iot network," *IEEE Access*, vol. 9, pp. 13938–13959, 2021.

[131] Y. Lu, "The blockchain: State-of-the-art and research challenges," *Journal of Industrial Information Integration*, vol. 15, pp. 80–90, 2019.

[132] A. S. Almasoud, F. K. Hussain, and O. K. Hussain, "Smart contracts for blockchain-based reputation systems: A systematic literature review," *Journal of Network and Computer Applications*, vol. 170, p. 102814, 2020.

[133] S. N. G. Gourisetti, M. Mylrea, and H. Patangia, "Evaluation and demonstration of blockchain applicability framework," *IEEE Transactions on Engineering Management*, vol. 67, no. 4, pp. 1142–1156, 2019.

[134] Y. Xu and Y. Huang, "Segment blockchain: A size reduced storage mechanism for blockchain," *IEEE Access*, vol. 8, pp. 17434–17441, 2020.

[135] W. Xiong and L. Xiong, "Data trading certification based on consortium blockchain and smart contracts," *IEEE Access*, 2020.

[136] O. Ali, A. Jaradat, A. Kulakli, and A. Abuhalimeh, "A comparative study: blockchain technology utilization benefits, challenges and functionalities," *IEEE Access*, vol. 9, pp. 12730–12749, 2021.

[137] H.-Y. Paik, X. Xu, H. D. Bandara, S. U. Lee, and S. K. Lo, "Analysis of data management in blockchain-based systems: From architecture to governance," *Ieee Access*, vol. 7, pp. 186091–186107, 2019.

[138] T. H.-J. Kim, L.-S. Huang, A. Perrig, C. Jackson, and V. Gligor, "Accountable key infrastructure (aki) a proposal for a public-key validation infrastructure," in *Proceedings of the 22nd international conference on World Wide Web*, pp. 679–690, 2013.

[139] D. Basin, C. Cremers, T. H.-J. Kim, A. Perrig, R. Sasse, and P. Szalachowski, "Arpki: Attack resilient public-key infrastructure," in *Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security*, pp. 382–393, 2014.

[140] J.-G. Dumas, P. Lafourcade, F. Melemedjian, J.-B. Orfila, and P. Thoniel, "Localpki: An interoperable and iot friendly pki," in *International Conference on E-Business and Telecommunications*, pp. 224–252, Springer, 2017.

[141] M. A. Vigil, C. T. Moecke, R. F. Custódio, and M. Volkamer, "The notary based pki," in *European Public Key Infrastructure Workshop*, pp. 85–97, Springer, 2012.

[142] R. Wang, J. He, C. Liu, Q. Li, W.-T. Tsai, and E. Deng, "A privacy-aware pki system based on permissioned blockchains," in *2018 IEEE 9th International Conference on Software Engineering and Service Science (ICSESS)*, pp. 928–931, IEEE, 2018.

[143] W. Wang, N. Hu, and X. Liu, "Blockcam: a blockchain-based cross-domain authentication model," in *2018 IEEE Third International Conference on Data Science in Cyberspace (DSC)*, pp. 896–901, IEEE, 2018.

[144] M. T. Hammi, P. Bellot, and A. Serhrouchni, "Bctrust: A decentralized authentication blockchain-based mechanism," in *2018 IEEE Wireless Communications and Networking Conference (WCNC)*, pp. 1–6, IEEE, 2018.

[145] A. Yakubov, W. Shbair, N. Khan, C. Medinger, J. Hilger, *et al.*, "Blockpgp: A blockchain-based framework for pgp key servers," *International Journal of Networking and Computing*, vol. 10, no. 1, pp. 1–24, 2020.

[146] L. Axon and M. Goldsmith, "Pb-pki: A privacy-aware blockchain-based pki," 2016.

[147] A. S. Ahmed and T. Aura, "Turning trust around: smart contract-assisted public key infrastructure," 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)*, pp. 104–111, IEEE, 2018.

[148] J. Chen, S. Yao, Q. Yuan, K. He, S. Ji, and R. Du, "Certchain: Public and efficient certificate audit based on blockchain for tls connections," in *IEEE INFOCOM 2018-IEEE Conference on Computer Communications*, pp. 2060–2068, IEEE, 2018.

[149] M. Y. Kubilay, M. S. Kiraz, and H. A. Mantar, "Certledger: A new pki model with certificate transparency based on blockchain," *Computers & Security*, vol. 85, pp. 333–352, 2019.

[150] M. Toorani and C. Gehrmann, "A decentralized dynamic pki based on blockchain," in *Proceedings of the 36th Annual ACM Symposium on Applied Computing*, pp. 1646–1655, 2021.

[151] S. Matsumoto and R. M. Reischuk, "Ikp: Turning a pki around with blockchains.," *IACR Cryptol. ePrint Arch.*, vol. 2016, p. 1018, 2016.

[152] B. Bünz, L. Kiffer, L. Luu, and M. Zamani, "Flyclient: Super-light clients for cryptocurrencies," in *2020 IEEE Symposium on Security and Privacy (SP)*, pp. 928–946, IEEE, 2020.

[153] L. Exosite, "Blockquick: Super-light client protocol for blockchain validation on constrained devices," 2019.

[154] E. Karaarslan and E. Adiguzel, "Blockchain based dns and pki solutions," *IEEE Communications Standards Magazine*, vol. 2, no. 3, pp. 52–57, 2018.

[155] E. F. Kfoury, D. Khoury, A. AlSabeh, J. Gomez, J. Crichigno, and E. Bou-Harb, "A blockchain-based method for decentralizing the acme protocol to enhance trust in pki," in *2020 43rd International Conference on Telecommunications and Signal Processing (TSP)*, pp. 461–465, IEEE, 2020.

[156] H. Zhang, X. Chen, X. Lan, H. Jin, and Q. Cao, "Btcas: A blockchain-based thoroughly cross-domain authentication scheme," *Journal of Information Security and Applications*, vol. 55, p. 102538, 2020.

[157] M. Conti, N. Dragoni, and V. Lesyk, "A survey of man in the middle attacks," *IEEE Communications Surveys & Tutorials*, vol. 18, no. 3, pp. 2027–2051, 2016.

[158] M. E. Hellman, "An overview of public key cryptography," *IEEE Communications Magazine*, vol. 40, no. 5, pp. 42–49, 2002.

[159] A. Alrawais, A. Alhothaily, X. Cheng, C. Hu, and J. Yu, "Secureguard: A certificate validation system in public key infrastructure," *IEEE Transactions on Vehicular Technology*, vol. 67, no. 6, pp. 5399–5408, 2018.

[160] A. P. Felt, R. W. Reeder, A. Ainslie, H. Harris, M. Walker, C. Thompson, M. E. Acer, E. Morant, and S. Consolvo, "Rethinking connection security indicators," in *Twelfth Symposium on Usable Privacy and Security (SOUPS 2016)*, pp. 1–14, 2016.

[161] E. Schechter, "Moving towards a more secure web," *Google Security Blog*, 2016.

[162] T. Vyas, "No more passwords over http, please," *Mozilla Blog*, 2016.

[163] R. Holz, J. Amann, O. Mehani, M. Wachs, and M. A. Kaafar, "Tls in the wild: An internet-wide analysis of tls-based protocols for electronic communication," *arXiv preprint arXiv:1511.00341*, 2015.

[164] S. Santesson, M. Myers, R. Ankney, A. Malpani, S. Galperin, and C. Adams, "X. 509 internet public key infrastructure online certificate status protocol-ocsp," tech. rep., 2013.

[165] S. Garfinkel, "Pretty good privacy (pgp)," in *Encyclopedia of Computer Science*, pp. 1421–1422, 2003.

[166] B. Laurie, A. Langley, and E. Kasper, "Certificate transparency," tech. rep., 2013.

[167] L. Chuat, P. Szalachowski, A. Perrig, B. Laurie, and E. Messeri, "Efficient gossip protocols for verifying the consistency of certificate logs," in *2015 IEEE Conference on Communications and Network Security (CNS)*, pp. 415–423, IEEE, 2015.

[168] Y. Wang and J. Vassileva, "Bayesian network trust model in peer-to-peer networks," in *International Workshop on Agents and P2P Computing*, pp. 23–34, Springer, 2003.

[169] B. Yu and M. P. Singh, "An evidential model of distributed reputation management," in *Proceedings of the first international joint conference on Autonomous Agents and Multiagent Systems: Part 1*, pp. 294–301, 2002.

[170] A. Yamamoto, D. Asahara, T. Itao, S. Tanaka, and T. Suda, "Distributed pagerank: a distributed reputation model for open peer-to-peer network," in *2004 International Symposium on Applications and the Internet Workshops. 2004 Workshops.*, pp. 389–394, IEEE, 2004.

[171] L. Xiong and L. Liu, "Peertrust: Supporting reputation-based trust for peer-to-peer electronic communities," *IEEE transactions on Knowledge and Data Engineering*, vol. 16, no. 7, pp. 843–857, 2004.

[172] H. Wen, X. Ren, and G. Xu, "A ds evidence theory based trust model for the p2p network," *Journal of Xi'An University*, vol. 32, no. 3, pp. 400–402, 2005.

[173] S. Song, K. Hwang, R. Zhou, and Y.-K. Kwok, "Trusted p2p transactions with fuzzy reputation aggregation," *IEEE Internet computing*, vol. 9, no. 6, pp. 24–34, 2005.

[174] R. Zhou and K. Hwang, "Powertrust: A robust and scalable reputation system for trusted peer-to-peer computing," *IEEE Transactions on parallel and distributed systems*, vol. 18, no. 4, pp. 460–473, 2007.

[175] Y. Sun, Q. Zhao, and P. Zhang, "Trust degree calculation method based on trust blockchain node," in *2019 IEEE International Conference on Service Operations and Logistics, and Informatics (SOLI)*, pp. 122–127, IEEE, 2019.

[176] J. Ahn, M. Park, H. Shin, and J. Paek, "A model for deriving trust and reputation on blockchain-based e-payment system," *Applied Sciences*, vol. 9, no. 24, p. 5362, 2019.

[177] W. She, Q. Liu, Z. Tian, J.-S. Chen, B. Wang, and W. Liu, "Blockchain trust model for malicious node detection in wireless sensor networks," *IEEE Access*, vol. 7, pp. 38947–38956, 2019.

[178] B. Zhao, Y. Liu, X. Li, J. Li, and J. Zou, "Trustblock: An adaptive trust evaluation of sdn network nodes based on double-layer blockchain," *PloS one*, vol. 15, no. 3, p. e0228844, 2020.

[179] Y. Inedjaren, M. Maachaoui, B. Zeddini, and J.-P. Barbot, "Blockchain-based distributed management system for trust in vanet," *Vehicular Communications*, vol. 30, p. 100350, 2021.

[180] F. Zola, M. Eguimendia, J. L. Bruse, and R. O. Urrutia, "Cascading machine learning to attack bitcoin anonymity," in *2019 IEEE International Conference on Blockchain (Blockchain)*, pp. 10–17, IEEE, 2019.

[181] S. S. Chawathe, "Clustering blockchain data," in *Clustering Methods for Big Data Analytics*, pp. 43–72, Springer, 2019.

[182] B. Huang, Z. Liu, J. Chen, A. Liu, Q. Liu, and Q. He, "Behavior pattern clustering in blockchain networks," *Multimedia Tools and Applications*, vol. 76, no. 19, pp. 20099–20110, 2017.

[183] D. Ermilov, M. Panov, and Y. Yanovich, "Automatic bitcoin address clustering," in *2017 16th IEEE International Conference on Machine Learning and Applications (ICMLA)*, pp. 461–466, IEEE, 2017.

[184] M. Harrigan and C. Fretter, "The unreasonable effectiveness of address clustering," in *2016 Intl IEEE Conferences on Ubiquitous Intelligence & Computing, Advanced and Trusted Computing, Scalable Computing and Communications, Cloud and Big Data Computing, Internet of People, and Smart World Congress (UIC/ATC/ScalCom/CBDCom/IoP/SmartWorld)*, pp. 368–373, IEEE, 2016.

[185] M. Fleder, M. S. Kester, and S. Pillai, "Bitcoin transaction graph analysis," *arXiv preprint arXiv:1502.01657*, 2015.

[186] A. Garba, Q. Hu, Z. Chen, and M. R. Asghar, "Bb-pki: blockchain-based public key infrastructure certificate management," in *2020 IEEE 22nd International Conference on High Performance Computing and Communications; IEEE 18th International Conference on Smart City; IEEE 6th International Conference on Data Science and Systems (HPCC/SmartCity/DSS)*, pp. 824–829, IEEE, 2020.

[187] L. Dykcik, L. Chuat, P. Szalachowski, and A. Perrig, "Blockpki: an automated, resilient, and transparent public-key infrastructure," in *2018 IEEE International Conference on Data Mining Workshops (ICDMW)*, pp. 105–114, IEEE, 2018.

[188] A. Yakubov, W. Shbair, A. Wallbom, D. Sanda, *et al.*, "A blockchain-based pki management framework," in *The First IEEE/IFIP International Workshop on Managing and Managed by Blockchain (Man2Block) colocated with IEEE/IFIP NOMS 2018, Tapei, Tawain 23-27 April 2018*, 2018.

[189] B. Qin, J. Huang, Q. Wang, X. Luo, B. Liang, and W. Shi, "Cecoin: A decentralized pki mitigating mitm attacks," *Future Generation Computer Systems*, vol. 107, pp. 805–815, 2020.

[190] H. Tewari, A. Hughes, S. Weber, and T. Barry, "X509cloud—framework for a ubiquitous pki," in *MILCOM 2017-2017 IEEE Military Communications Conference (MILCOM)*, pp. 225–230, IEEE, 2017.

[191] T. S. Madhulatha, "An overview on clustering methods," *arXiv preprint arXiv:1205.1117*, 2012.

[192] K. P. Sinaga and M.-S. Yang, "Unsupervised k-means clustering algorithm," *IEEE access*, vol. 8, pp. 80716–80727, 2020.

[193] J. Baarsch and M. E. Celebi, "Investigation of internal validity measures for k-means clustering," in *Proceedings of the international multiconference of engineers and computer scientists*, vol. 1, pp. 14–16, sn, 2012.

[194] K. Khan, S. U. Rehman, K. Aziz, S. Fong, and S. Sarasvady, "Dbscan: Past, present and future," in *The fifth international conference on the applications of digital information and web technologies (ICADIWT 2014)*, pp. 232–238, IEEE, 2014.

[195] N. Truong, G. M. Lee, K. Sun, F. Guitton, and Y. Guo, "A blockchain-based trust system for decentralised applications: When trustless needs trust," *Future Generation Computer Systems*, vol. 124, pp. 68–79, 2021.

[196] W. Hao, J. Zeng, X. Dai, J. Xiao, Q.-S. Hua, H. Chen, K.-C. Li, and H. Jin, "Towards a trust-enhanced blockchain p2p topology for enabling fast and reliable broadcast," *IEEE Transactions on Network and Service Management*, vol. 17, no. 2, pp. 904–917, 2020.

[197] Y. Zhang, F. Tong, Y. Xu, J. Tao, and G. Cheng, "A privacy-preserving authentication scheme for vanets based on consortium blockchain," in *2020 IEEE 92nd Vehicular Technology Conference (VTC2020-Fall)*, pp. 1–6, 2020.

[198] D. Boughaci and A. A. Alkhawaldeh, "Enhancing the security of financial transactions in blockchain by using machine learning techniques: towards a sophisticated security tool for banking and finance," in *2020 First International Conference of Smart Systems and Emerging Technologies (SMARTTECH)*, pp. 110–115, 2020.

[199] C. Udokwu, H. Anyanka, and A. Norta, "Evaluation of approaches for designing and developing decentralized applications on blockchain," in *Proceedings of the 2020 4th international conference on algorithms, computing and systems*, pp. 55–62, 2020.

[200] C. Pop, T. Cioara, I. Anghel, M. Antal, and I. Salomie, "Blockchain based decentralized applications: Technology review and development guidelines," *arXiv preprint arXiv:2003.07131*, 2020.

[201] T. C. Dao, B. M. Nguyen, and B. L. Do, "Challenges and strategies for developing decentralized applications based on blockchain technology," in *Advanced Information Networking and Applications: Proceedings of the 33rd International Conference on Advanced Information Networking and Applications (AINA-2019) 33*, pp. 952–962, Springer, 2020.

[202] C. Patsonakis, K. Samari, A. Kiayias, and M. Roussopoulos, "Implementing a smart contract pki," *IEEE Transactions on Engineering Management*, vol. 67, no. 4, pp. 1425–1443, 2020.

[203] Y. C. E. Adja, B. Hammi, A. Serhrouchni, and S. Zeadally, "A blockchain-based certificate revocation management and status verification system," *Computers & Security*, vol. 104, p. 102209, 2021.

[204] S. S. Gupta, "Blockchain," *IBM Onlone (http://www. IBM. COM)*, 2017.

[205] G. Ethereum, "Go ethereum," 2020.