Blockchain-based Security Model for Efficient Data Transmission and Storage in Cloudlet Network Resource Environment

by

Nothile Clementine Masango

Student No.: 220081146

A dissertation submitted to

the University of KwaZulu-Natal,

College of Agriculture, Engineering and Science,

in fulfilment of the requirements for the degree of

Master of Computer Science

in the

School of Mathematics, Statistics and Computer Science

University of KwaZulu-Natal

Durban, South Africa

March 16, 2023

Declaration - Authorship

I, **Nothile Clementine Masango**, declare that this dissertation titled, **'Blockchain-based security model for efficient data transmission and storage in cloudlet network resource environment'**, and the work presented in it is my own. I confirm that:

1. The research reported in this dissertation, except where otherwise indicated, is my original research.

2. This dissertation has not been submitted for any degree or examination at any other university.

3. This dissertation does not contain other persons' data, pictures, graphs or other information, unless specifically acknowledged as being sourced from other persons.

4. This dissertation does not contain other persons' writing, unless specifically acknowledged as being sourced from other researchers. Where other written sources have been quoted, then:

    - Their words have been re-written but the general information attributed to them has been referenced.
    - Where their exact words have been used, then their writing has been placed in italics and inside quotation marks, and referenced.

5. This dissertation does not contain text, graphics or tables copied and pasted from the Internet, unless specifically acknowledged, and the source being detailed in the dissertation and in the references sections.

Nothile C. Masango

DECLARATION – SUPERVISOR

As the candidate's supervisor, I agree to the submission of this dissertation.

Supervisor: Prof. Absalom Ezugwu

Abstract

As mobile users' service requirement increases, applications such as online games, virtual reality, and augmented reality demand for more computation power. However, the current design of mobile devices and their associated innovations cannot accommodate such applications because of the limitations they have in terms storage, computing power and battery life. Therefore, as a result, mobile devices offload their tasks to the remote cloud environments. Moreover, due to the architecture of cloud computing, where cloud is located at the core of the network, applications experiences challenges such as latency. This is a disadvantage to real-time online applications. Hence, the edge computing based cloudlet environment was introduced to bring resources closer to the end user, with an enhanced network quality of service. Although there is merit in deploying cloudlets at the edge of the network, which is closer to the user, this makes them susceptible to attacks. For this newly introduced technology to be fully adopted, effective security measures need to be incorporated into the current cloudlets computing platform. This study proposes blockchain technology as a security model in securing the data shared between mobile devices and cloudlet, with an agent layer concept introduced in between mobile device layer and cloudlet. The im-

plemented agent-based model uses the new consensus mechanism, proof of trust where trust and experience is determine by the number of coins each node (cloudlet) possess, to select two miners. These miners participate in message verification using Elliptic curve scheme, and if they do not reach consensus, a third miner is selected to resolve the conflict. Any miner with wrong verification loses all the coins; in this way trust and experience is controlled. This proposed solution has proven to be more efficient in terms of security and network performance in comparison to existing state-of-the-arts implementations.

Acknowledgements

To God, the father who has been with me since the beginning of this journey and made the completion of this dissertation possible.

To my late grandmother, for raising me to be the strongest woman I am today. Without her teachings, I would not have been able to overcome challenges I faced while working on this dissertation.

To my supervisor, Prof. Absalom Ezugwu, I cannot thank you enough for your support, patience, guidance, and advice throughout the journey. I am forever grateful.

Lastly, the financial support from Council for Scientific and Industrial Research (CSIR) towards this research is duly acknowledged.

Contents

List of Figures

List of Tables

Chapter 1

1.1 Introduction

In recent years, the number of mobile devices such as smart phones and tablets has increased drastically[3]. The advanced network technology such as 5G and LTE has given birth to revolutionary mobile application frameworks. These applications are resource-demanding, requires more space for data storage, and more power for intensive computation. More so, mobile devices are too resource constrained to accommodate such applications [1].

To support lot of users and expand services with limited resources, cloud computing was introduced. Cloud computing offers high processing power to many applications on the network, as well as memory and storage to meet users forever changing and advancing needs [4]. The introduction of this paradigm has enabled the implementation of innovative applications without worrying about mobile devices limitations. By migrating data or computation to cloud server, mobile devices can use cloud server as an extension of themselves [1]. Cloud computing offers services that brings flexibility and

mobility in the industrial environment. These services allow mobile devices to offload the resource-demanding task to the cloud to do the computations and return the results to the mobile user. However, due to the geographical distance between the cloud server and the users, this might result in communication delays and bandwidth cost when many devices are connected to the internet. Latency-sensitive applications such as real-time applications may suffer due to complex network conditions in wide area network (WAN) environment [1].

In recent development, the cloudlet paradigms attempt to overcome the above-mentioned challenges.The cloudlet technology offloads some computation tasks and storage from cloud to the edge devices and vice versa. However, this raises many concerns regarding security. Also, all the possibility brought by edge computing such as distributed cloudlets, and, huge data processing, has resulted in the existing security mechanisms not being effective enough to secure edge computing network [5].

Several researchers have adopted the concept of blockchain technology to provide security in the three tier architecture. A blockchain is a distributed database or ledger that is shared among the nodes of a computer network. This technology stores all electronic transaction from nodes in the order it occurred [6]. Furthermore, the innovation with a blockchain is that it guarantees the fidelity and security of a record of data and generates trust without the need for a trusted third party. Transactions are duplicated and distributed among nodes present on the network. In blockchain each block is

linked to the previous block using cryptography. When a block is added to the chain, the copy is distributed among other nodes present in the network for validation. This makes it difficult for the data in blockchain to be modified, hence blockchain is consider as one of the most reliable security mechanism compared to the other existing state-of-the-arts related technologies. However, most of them are found unfit due to high energy consumption [7] and single point of trust[3]. Hence this study proposes data integrity enabled configuration using blockchain technology, with a reduced power consumption and controlled trust concept using blockchain.

1.1.1 Problem statement

The cloudlet computing paradigm has brought so many technological initiatives and capabilities such as offloading computes intensive resource applications. However, despite this advantage, users have no control over their private information, and more so, in this computing environments, data and information are susceptible to threats such as data loss, data modification, etcetera. This raises concerns and calls for the need to employ mechanisms which will ensure data integrity and monitor data modification when data is in transit and when it is stored. Moreover, blockchain technology has been previously introduced in the literature, but the current proposed methods consume more energy which affects the performance of the entire network and thus in turn affects security. In this study, new security enhancements and energy efficient mechanisms are implemented to address the current data transaction and information storage challenges in the cloudlets' environments.

1.1.2 Motivation

Despite all the benefits of cloudlet technology, security is still a major challenge, and since cloudlet is the linkage gateway in terms of connection from mobile devices to the cloud, security measures need to be implemented since most user have no control over their data once it has been offloaded. Several authors discuss some of the security and privacy issues in the cloudlets computing environment [8].

Cloudlets are deployed at the edge of the network, which is closer to the user. This makes them susceptible to attacks such as man-in-the-middle, IP address spoofing, data tempering. Since the nature of cloudlet computing is to offload certain storage and computation task from cloud data centres to the edge of the network and vice versa. This raises many concerns in terms of security and privacy. One of these challenges is of data confidentiality. In this paradigm data from users is outsourced to the edge server and user loses control over their private data. This sensitive data is prone to threat such as data loss, and data breach. Thus, a security mechanism is required to ensure that data is protected when outsourced to the edge server. Another challenge is the issue of data integrity: in transit data can be tempered with by advisories and hence raises the need for a trusted auditor entity.

In cloudlet computing environment, when offloading occurs, the technology is unable to distinguish normal devices from attackers. As a result, security of transmitted data cannot be assured and users' privacy may be compromised as well [9]. Also, the forever increasing user's requirements of task migration

has resulted in the challenge of ensuring data integrity during offloading and execution [10]. Thus a trusted auditor entity required to oversee the data modification before and after storage in the cloudlet.

Authors [9] and [11] have adopted the blockchain technology to secure cloudlet network. In their studies they adopted proof-of-work (PoW) consensus mechanism, where all nodes on the network take part in validating the block before being added on the chain. However, adopting such mechanism is not ideal as it requires intensive power to solve some of the mathematical puzzles that might arise afterwards, coupled with the fact that the delay caused by the mining process is not suitable for real-time applications execution [3]. More so, the energy consumption during transmission of a task in cloudlet computing is still a big problem that needs to be solved [10]. In addition, while security is considered very important, the quality of service (QoS) is also vital: achieving one without the other is not an advisable situation.

1.1.3 Aim and Objectives of the Study

Recent studies have shown that the increasing users' requirements for migrating tasks pose a challenge to preserving the security and integrity of offloaded data processed by cloudlets. Therefore, based on this notion, the study aims to implement the integration of blockchain based security solutions to ensure several security services such as transaction traceability and secure exchanges between the user cloudlets and the cloud servers. Furthermore, due to the lack of security and data synchronization problems in cloud systems caused by some deadlock situations, an agent-based model is integrated into the

proposed framework to control and resolve conflicts concerning transactions among cloudlets. To test and evaluate the feasibility of the study, the aforementioned security schemes will be applied on response-sensitive applications and data modification attack will be launched. The overall goal of this research has been broken down into the following specific objectives:

- To investigate the state-of-the-art literature on an existing cloudlet security model.
- To investigate a suitable blockchain technology for securing cloudlet.
- To design and implement a blockchain model in the cloudlet network that would improve the security of data whilst minimizing energy consumption.
- To evaluate the performance of the developed security model using response-sensitive applications.

1.1.4 Contributions

The contributions of this research to knowledge and humanity are as follows:

1. We designed and implemented a blockchain model in the cloudlet network that improves the security of data in transit whilst minimizing energy consumption.

2. We evaluated the performance of the developed security model using response-sensitive applications to show the viability of the proposed security models.

3. We adopted some of the trust concept, reputation and experience from [12], which will be measured by the number of coins each node has. This concept was employed and used in the proposed agent-based scheme in selecting two nodes that will validate the blockchain block.

4. Finally, we carried out extensive comparison of the two nodes' results and include another two nodes selected by the agent. So, if any node produces wrong result, that node loses all the coins.

1.1.5 Outline

- Chapter 2 - In this chapter, an overview of cloudlet computing as well as blockchain was discussed. Literature related to the study was analyzed.

- Chapter 3 - This chapter explained the research methodology used to achieve the objectives outlined in chapter one of this thesis.

- Chapter 4 - In this chapter, a detailed explanation of how the project implemented was given. This section also discussed of the results of proposed security mechanism and how it relates to previous work and the objectives of this research.

- Chapter 5 - This chapter gives a summary of proposed work. Suggestions on providing possible improvements were included in this chapter.

Chapter 2

2.1 Overview of cloudlet

2.1.1 Cloud computing

In recent years mobile applications as well as systems have emerged. Cloud computing's integration with real-time data and web applications has resulted in drastically increase in all categories which includes entertainment, health, social networks, games and businesses. All these advancement puts pressure on mobile devices as well as Internet of Things (IoT) devices as they have limited resources - battery life, storage capacity, process power [13]. Authors in [14] proposed cloud computing as a solution to the shortfalls of mobile and IoT devices. Cloud computing offers services that bring flexibility and mobility in the industrial environment. These services allow mobile and IoT devices to offload the resource-demanding task to the cloud to do computation and return results to the mobile users as well as data storage is hosted on cloud then on the device itself. It also provides easy access on storage and computation resource on demand. Cloud computing

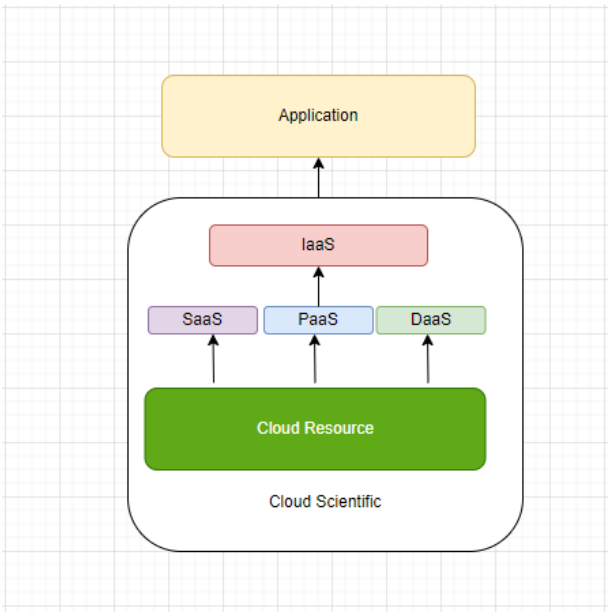


Figure 2.1: Cloud computing architecture

architecture has services which are used to access computing resource on the cloud over the internet shown in figure 2.1 .This serves as a mechanism to handle forever changing customers' requirements and demands[15]:

- Software as a service (SaaS) - Cloud consumers host software using cloud computing resources which can be accessed by application users through different clients (web, browsers). In SaaS, a user has no control over the cloud infrastructure. Servers, storage, network, required for hosting the software, are not managed by users but the cloud provider. Rockspace, saleforce.com are examples of this service.

- Platform as a service (PaaS) - It allows cloud consumers to develop and host their application directly on the cloud. PaaS hosts both completed and in-progress applications. Microsoft Azure, google app engine are

examples of this service.

- Infrastructure as a service (IaaS) - Cloud consumers use all the infrastructure offered by the cloud, directly on the cloud. Such infrastructure includes processing, storage, networks. Amazon EC2, Amazon s3, Sunnis cloud services are examples of IaaS.

- Data storage as a service (DaaS) - Data storage service is the delivery of virtual storage on demand. DaaS minimize cost by only allowing consumers to pay for what they use only, than to pay for the entire database offered in the cloud.

Components of cloud computing

- Hypervisor - A hardware that is a virtual machine manager that allow users to create and run VMs on a single hardware host. It also manages different Operating Systems which can make use of virtualized resource of hardware.

- Virtualization - It is used to merge resources (network, storage, Operating System) to virtual environment. This environment provides many benefits such as reduced hardware costs and enhanced reliable performance.

- Storage - Data is stored on cloud over the network. This data is backed-up, managed and maintained remotely. Benefits of using cloud storage are as follows: information management, pay as you use, as well as time deployment.

- Multi-tenancy - This environment consists of a single application software that can serve multiple users. Only the application services can be shared not user's data. Users are allowed to customize the front-end of the application but not the back-end.

Advantages of cloud computing

Moving to the cloud came with a lot of advantages[16]. Organisation are able to utilise their infrastructure fully and increase utilisation of servers with the use of cloud computing.

- Reliability - With cloud computing, users can access data in any location. This allows applications utilising cloud services to be more reliable.

- Better storage and back-up capacity - Cloud provides virtual storage, enough for organisations to use. This storage is flexible enough to meet users ever-changing demands.

- On-demand self-service - The users have unlimited availability of computing resources for their products. These resources include CPU time, storage, software use, which can be accessed automatically with no further interaction from any human.

Challenges in cloud computing

Apart from all the advantages the cloud computing brings it still has some drawbacks. Some of the challenges are as follows:

- Security - Cloud computing suffers from security issues such as data loss, phishing and botnet during offloading. These issues pose a serious threat to organizations' data. As data is accessed from any location, it poses a threat to users privacy.

- Costing Model - As much as migrating to cloud reduces cost in terms of infrastructure but the cost of data communication increases - communication from organization to the central cloud.[13]

- Network performance - To overcome security and ensure data security confidential data may be split and then send to cloud and this affect the system's performance . Transportation of huge amounts of data generated at the edge is becoming a bottleneck to the cloud computing paradigm, resulting to high latency which affects performance of real-time applications [17].

- Single point of failure - Cloud becomes one controller of the whole network, hence if it fails, the whole system will fail[18].

Existing mechanisms for minimizing the energy consumption in cloud computing are unable to meet the increasing demand for data energy consumption. Due to the forever improving technologies, higher requirements for energy consumption are highly expected[19].

Edge computing : cloudlet

To resolve the above-mentioned cloud challenges, such as network performance, and single point of failure, edge computing was introduced. Edge computing is a new paradigm that brings cloud resources at the edge of the

network. According to [20], edge computing is a new computing model that deploys computing and storage resources(cloudlets, fog nodes, etc) at the edge of the network, closer to mobile devices or sensors. Edge computing will never replace cloud computing, these two technologies should co-exist. After data has been processed at the edge of the network it is further uploaded to cloud. In some cases, data nodes still need further processing in the cloud for meaningful results analysis [19].

A cloudlet is one of the edge computing model, which is a collection of computers that are connected to the internet, bringing resource at the edge of the network closer to the user[21], and meet challenges of the mobile cloud computing. It is located 1(one) hop away from the mobile user. Because of computation offloading service in cloudlet, performance is improved, and power consumption is reduced on applications in resource-constrained devices.

Cloudlet utilities

Cloudlet technology is 3-tiered architecture of mobile device-cloudlet-cloud, as shown in Figure 2.2. In this technology, mobile devices send jobs to the cloudlets for required processing and return the final results: this process is called offloading. Offloading data to the cloudlet has brought so many possibilities and advancement in the IT sector, without having to worry about inabilities of mobile devices. There are 2 types of offloading:
1. Cloudlet-based computation offloading: In this technology, mobile devices upload resources demanding tasks to available close cloudlet for computa-

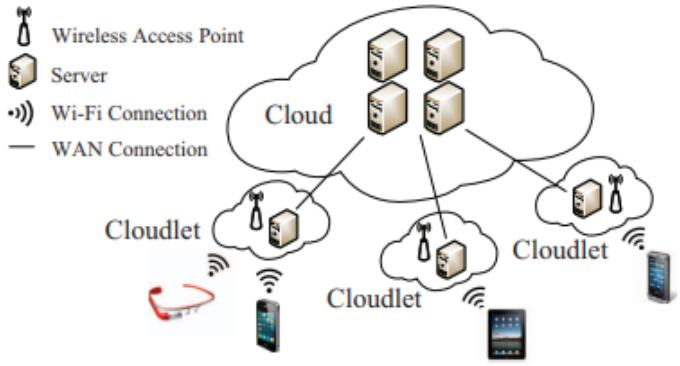


Figure 2.2: Cloudlet architecture [1]

tion. In this instance, cloudlet is said to be a data center in a box whose main aim is to bring cloud closer to the user. Face recognition, augmented reality, crowd-sourcing video processing are examples of this type offloading [1].

2. Cloudlet-based data offloading: In this technology, data is offloaded to cloudlet for storage. Data is cached in cloudlet to improve data transfer performance between mobile devices and cloud [1]. Video on demand,cloud storage, video surveillance are some of the applications of this type of offloading. Lowering communication latency, and improving connectivity are some of the utilities offered by cloudlet [5]:

- Rapid response: Technology advancement has brought possibilities of intensive storage-applications which are resource hungry. These applications require fast response. Moreover, cloudlet can accommodate such requirements through offloading computation and storage on cloudlet which is closer to the user, hence application response is improved. The WLAN link is usually used for communication between

mobile devices and cloudlets. This link has a higher bandwidth capacity, resulting to the performance being improved [22]

- Cloudlet outage control: Service outage affects the public confidence in Information Technology industry. VM-based cloudlet overcomes the unavailability of the cloud and only synchronise data when the cloud comes back to live.Cloudlet services can be utilised anytime as it connection does not depend on the availability of internet connectivity [22].

- Last-mile security: Cloudlet is the first point of connection to the user, where security policies can be deployed before data is transferred to the cloud.This ensures that the data reaching the cloud is secured.

2.2 Overview of blockchain

Blockchain is a data structure that is decentralized containing transactions between multiple parties. In [23] they define blockchain as immutable ledger which allow transactions to take place in a decentralized manner. It uses consensus mechanism to verify every transaction before adding it to the distributed ledger. Blockchain has been adopted in many systems to ensure security because once the details of the transaction are added in the chain they cannot be modified. Each block is built on top of the other, as it carries its own hash, and the hash key of the previous block. In this way data cannot be modified; the hash key changes and data tempering can be easily detected. The blockchain technology is applicable to IoT, systems that prioritises reputation, security services, financial services such online payments,

and business that needs to ensure reliability and honest.

Integrating blockchain in edge computing bring possibilities of reliable access, data storage at the edge of the network in a secured manner. The blockchain immutability enhance data integrity in edge computing. When messages are transmitted, they are susceptible to attacks. Thus, network administration needs to be trustworthy and validated. Blockchain offers network administration without the third part interference, and enables each node to manage and control access on its own data without any outsider.

2.2.1 Blockchain architecture

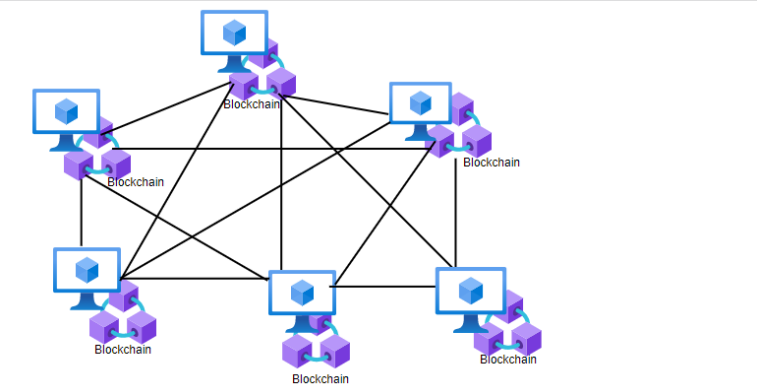


Figure 2.3: Blockchain architecture

Blockchain technology is a peer-to-peer architecture, shown in figure 2.3. It is made up of nodes, called miners, that create a block, and use cryptography to join them to form a chain. The process of creating a block is called mining. Block is made up of header and the body. Moreover, it consists of transaction counter, hash value of all transaction in the block, nonce (4-byte

field that start with zero), and a set of block validation rules. When these blocks are linked together via hash keys, they form a blockchain.

Consensus mechanism

Consensus mechanisms are used as a fault-tolerant mechanism for verifying transactions in blockchain. Consensus ensures that an agreement is reached among nodes in a network. Different types of consensus mechanism exist in blockchain technology.

Proof of work

Bitcoin network is one of the systems that adopted this consensus mechanism. In this mechanism, each node has to find the hash value of the block header through changing a nonce. It uses nonce to calculate the hash key and broadcast the block to other nodes for verification of the hash value. However, since all the nodes take part in mining it is highly likely that 2 miners can mine the block simultaneously. But in Proof of work (PoW) a longer chain is said to be the authentic one. Mining in this mechanism require miners to have many computers to calculate and that is a waste of resources.

Proof of stake

The proof of stake is a solution to consumption issue in PoW [23]. In this mechanism it is assumed that rich nodes or people are not likely to attack the network. The selection of the miner is based on bank balance of the individuals. Selecting based on account balance is not fair as a single richest person can be dominant in a network.

Delegated proof of stake

The delegated proof of stake (DOPOS) is representative democratic while PoS is direct democratic [24]. In this mechanism, stakeholders select a miner to mine the block and also validate the block. Since few nodes will be involved, validation process will be quick. Additionally, if there is any suspicion on dishonest of delegation,they are voted out immediately.

2.2.2 Advantages of blockchain

- Immutable: Once a transaction is added on the blockchain it cannot be deleted or modified.
- Irreversible: According to [25], this feature "prevents double spending".
- Distributed system: All the members in the blockchain network are in position of a copy of the ledger.
- No centralized authority: It is a peer-to-peer system; no central server controlling the blockchain network or any third party is needed to validate each transaction.
- Persistence: Rollback or deletion is not possible in blockchain.
- Anonymity: Users can interact with the blockchain without their identities being revealed.

2.3 Security in cloudlets

The truth remains unchanged that cloudlet brought many possibilities that resolved mobile cloud computing drawbacks. However, cloudlet still suffer from the security threats like cloud computing and more. There are risks to the data stored in the cloud, anyone can access and modify such data. Hence data needs to be protected from malicious users. Authentication, privacy, confidentiality, and data security are still major security issues in cloudlet computing [13].

Cloudlets are not trusted entities; offloading mobile device's workload to a local cloudlet for computation is a major issue when it comes to security and privacy [26]. Cloudlets need to be authenticated before offloading could take place. Apart from offloading to not trusted cloudlet, offloading over the edge causes risks as data might be exposed to malicious users while in transit, such attacks are known as eavesdropping,and data modification. In addition, users have no control over their data once it is in process of offloading: any attack performed on data might take time to be discovered.

Security and privacy are a major issue in personal data especially when shared and accessed over wireless networks. Existing data protection protocols are suitable for the cloudlet technology, and benefits of cloudlets are meaningless if proper security and privacy protocol are not put in place to resolve damages caused by malicious users. Data security and privacy in cloudlet computing remain unsolved [26].

2.4 Related work

Once data has been offloaded on cloudlets, users have no control on what happens to it. This raises a need for security measures that will ensure that data is secured since most of the users' data are private. Jindel and Dave *et al.* [13] proposed a data security protocol that adopts the concept of perfect forward secrecy concept. The mobile user upload encrypted data with session key, generated using Diffie-Hellman key exchange algorithm to cloudlet and have full control of who accesses the file. The cloudlet caches the encrypted file and forward it to the cloud.If another user wants to access the file, it sends a request and its own ID to the owner of data for verification. While waiting for verification, its sends request to access data to the nearest cloudlet. The cloudlet gets the requested file either from its cache or from the cloud and forward it to the user. Once the data owner has authorised the request, the owner downloads the file session key, decrypt it and encrypt with data requester ID. Now the user can decrypt the file and have full access on data. However, users can get the file from cloudlet before being authorised by the data owner. This might be too risky especially when the session key is leaked, the malicious will have access to the private user data.

Because of the blockchain advantages such as security and traceability, many researchers have adopted this technology as security mechanism. Authors in [6] proposed a blockchain technology to secure exchange of data between cloudlet and cloud server, and traceability of transaction. The cloud divides the task to be processed into sub-tasks and send them to cloudlets.The transactions between the cloudlets are then grouped into blocks, validated

using public cryptography, and added on the blockchain. This blockchain is distributed to all the entities on the network for verification and validation. They also proposed a super-agent component to resolve drawbacks due to the lack of synchronization in the cloud technology. The super-agent component selects cloudlets to process the task from the cloud. This agent authenticates and records newly joining nodes on the network.

To ensure integrity on data offloaded in the cloud, Xu *et al.* [9] proposed a blockchain-based cloudlet management method of multimedia workflow to secure data during offloading. Cloudlets compete for recording offloaded data in a block by performing proof-of-work. Once the block has been mined by a winner, it is distributed to other nodes for verification. Once verified, it is added to the chain and objective cloudlet processes the data, or else migration is cancelled. In [27] they utilised consortium blockchain and DSSC and ISSC smart contract technologies to ensure data storage and sharing is secured in vehicular edge network. They also adopted elliptic curve digital signature algorithm and asymmetric cryptography to authenticate vehicles. For consensus, PoW was used and is processed in vehicular edge nodes. However, Xiong *et al.* [7] propose edge concept for blockchain technology and resource management approach to resolve the issues that arises from the blockchain consensus mechanism, referred to as proof-of-work. This requires nodes to solve mathematical puzzles for block validation which needs lot of resources such as computing and storage. Since the mobile devices have limited resources,it offload the proof-of-work to be computed by the edge servers. To overcome the resource allocation issues, miners (mobile devices) are charged

by the providers (edge servers). The providers increases pricing based on demand. Hence miners need to consider both the price and rewards to be gained before requesting the providers service. However, delay due to wireless is neglected and small network is used in this study.

The existing blockchain concepts cannot be directly applied to the edge network. They require a lot of resources for computation when mining the block. This raises a need for new consensus protocols accommodating specific characteristic of edge nodes and devices, as well as ways of reducing energy consumption in blockchain and edge computing integration. Thus, l-Mamun and Zhao *et al.* [28] proposed a new consensus mechanism which matches the characteristics of edge nodes and edge devices is proposed. Decentralized Edge Autonomous Network (DEAN) is lightweight consensus protocol that "leverage the resources" available in the edge nodes to decrease the pressure on edge devices. They are reliable to support fast processing in edge computing, and allow data sharing. This mechanism is built based on four protocols which includes distribute network construction that deals with the selection of the leader by edge nodes based on votes, DEAN Consensus where leaders are the only entities involved in block validation, validation of new nodes joining the network by leaders, and distribution of data and failure recovery.

Qiao *et al.* [29] propose a device-to-device edge computing and network framework that uses blockchain to address security challenges due to lack of trustworthiness between task owners and resource management during computational offloading. This framework facilitates low-latency execution in

real-time IoT applications. In their blockchain concept, they present a new proof-of-reputation consensus mechanism (PoR) where a miner is elected based on reputation score. This score is determined based on computation performance and reputation history. Task owners are responsible for rating reputation of service providers after computation offloading process has been completed.

Moreover, Jayasinghe *et al.* [3] proposed privacy preserving blockchain called Trustchain which integrate blockchain with edge computing and adapt trust concept to get rid of privacy issues associated with traditional blockchain.In this study, trust is used to define consensus for block validation in a network. In contrast to existing consensus mechanism that select a validator based on wealth, or computation power. In Trustchain, a miner is selected based on the level of trustworthiness and each miner gives consent to possible miners.

In [30], a distributed and trusted authentication for edge computing is proposed based on blockchain.They adopted Byzantine Fault Tolerance Consensus algorithm for storing authentication data and logs to ensure traceability. A leader is selected using round robin algorithm instead of computing complex mathematical puzzles.This consensus algorithm is executed for verifying identity and storing authentication logs in blockchain, to achieve data traceability.

Facilitating trust in a decentralized manner in edge computing is still a challenge. Bonnah *et al.* [31] proposed a fully decentralised approach to solve

scalability issues that come from single trusted entity, used for authentication between edge servers and users by eliminating public trusted entity. During authentication, the users public key is broadcast to all the nodes on the network for validation, and when accepted, credentials for all nodes is shared with the user. Thus user is authenticated once and access all the services/resources on the same network. All the user' keys are stored in blockchain and distributed to edge servers.

However, in evaluating the effectiveness of their proposed method, authors did not pay much attention on the impact of the computational resource on mobile user devices. Furthermore, if one of the users details are compromised, the entire security will be compromised. Giving away all credentials on one validated user results in single point of attack. Moreover, Yuan *et al.* [32] proposed a blockchain-based decentralised platform to support edge computing called CoopEdge. In their study, respective edge server publishes computation task to other edge servers to compete on. Performance history recorded on blockchain, and current latency is used to determine the winner for peer-task offloading and granting reward, as well as for selection of the miner. However, in their study, they investigated performance (time taken for peer offloaded task to an edge, CPU utilization, consensus) but no possible attacks against their platform were investigated, which makes it hard to determine whether their platform is secured enough.

During task offloading, transmitted information is susceptible to attacks which could result in data deficiency. Xu *et al.* [33] proposed a blockchain en-

abled computation offloading method, referred to as BeCome. This method ensures data integrity in edge computing, aiming to decrease time taken to offload task and energy consumption of edge computing tasks to achieve load balancing in IoT systems. In their study, they used genetic algorithm to generate balance resource allocations.

Security of data storage under edge computing remains one of the major issues. To solve such issues, the authors in [34] proposed a blockchain combined with regeneration coding to improve security and reliability of stored data under edge computing. Ren *et al.* proposed hybrid storage architecture and model under edge computing. Redundancy is introduced to improve system reliability. The proposed scheme consists of two types of blockchain, local and global blockchain. Local blockchain is created by edge servers to store data collected by IoT devices. In global blockchain, data on local blockchain is periodically uploaded to the global blockchain which is in the cloud server. The hashes of data are periodically validated to ensure data integrity comparing hashes from local and global. A private key is used to sign the hash key of data block and validating edge devices.

Moreover, in [35], authors present a blockchain-based trusted data management scheme, named BlockTDM. BlockTDM includes mutual authentication protocol, consensus that is flexible to use, blockchain nodes management. In their study, blockchain is executed on edge nodes. Smart contract was used for decryption of blockchain transaction, Multi-signature was adopted for reaching consensus ,and mutual authentication for encrypting sensitive data

using certificates before adding it to blockchain. However, cloud is responsible for complex problem solving which might results to latency and affect the network performance.In addition, only two nodes were used in experimental work.

To gain trust on cloudlet during offloading, they need to be authenticated. In [36] a security protocol is presented to authenticate cloudlet using mobile devices. Their proposed solution consists of secure element which stores crucial information such as pin and security keys, NFC-enabled mobile application, mobile network operator (MNO). To initiate the authentication process, mobile devices send a discovery message to cloudlet based on distance and other security criteria. Once cloudlet has been discovered, a private key from secure element is used to sign the application token to authenticate the cloudlet. The cloudlet sends its identity and private key to trust a service manager. TSM generates a signature and establish communication with MNO for a secure channel.

According to Arif *et al.*, [37] task computation is not done securely due to unpredictable task arrival. Hence, they proposed a secure and energy-efficient computation offloading scheme of mobile device using LSTM algorithm. The predictions of computational tasks obtained from the LSTM algorithm were used for the strategy of computational offloading of mobile devices. LSTM serves as a firewall that protect user devices.

Compromised edge devices results in to distributed denial-of-service (DDOS).

Hence, Bhardwaj *et al.* [38] proposed a new approach in which a function is deployed at the edge of the network to collect necessary information about incoming traffic in edge computing network. This scheme assists in detecting, arresting of attacks such DDOS, and limiting their malicious impact.

2.5 Summary

This chapter introduce the background of cloudlet which is one of the edge computing implementations. Edge computing brought many possibilities including solutions to limitations of the cloud, namely latency, reduced resource consumption, etc. New features in edge computing, including large number of technologies, new applications as well network at the edge have raised many security concerns. Moreover, real-time applications that have adopted this paradigm, have discovered performance as being being another important aspect in edge computing. Hence, for a reliable edge computing system, security and performance should go together.

Many authors have integrated blockchain technology with edge computing as a solution to enhance security. However, in their proposed schemes, PoW consensus mechanism was adopted, which results in high energy consumption due to complex mathematical problem solving, thereby affecting the performance of the network. Moreover, some authors adopted Proof of reputation as a consensus mechanism in which trust is put to a single node. If this node is compromised,it will not be detectable, especially when it has no competition. It always passes the reputation test due to its riches or processing power. Furthermore, most mechanisms do not authenticate the edge devices which may results to attacks like denial of service. Table 2.1 gives a summary of merit and demerit for certain approaches

Hence, this study proposes blockchain as a security mechanism to secure data in cloudlets. The study also proposes an agent layer between the edge

devices layer and cloudlet layer for authenticating both cloudlets and edge devices and adopt the concept of trust [39] where we determine trust based on experience and reputation judging by the number of coins in each node's position. A proof of stake is adopted but with two miners instead of one where each miner is determined by the results from the trust concept. To balance the reputation,coins are taken away from the miner on false validation.

Table 2.1: Merit and demerit of related work approaches

Ref	**Merit**	**Demerit**
[28]	DEAN is lightweight consensus protocol that decreases the pressure on edge devices, are reliable to support fast processing in edge computing, and allow data sharing. This consensus mechanism has a reduced energy consumption.	A single node is trusted with mining with no other validation Putting trust to single node increases security risks.
[31]	Fully decentralized approach that solve scalability issue eliminating the need of public trusted entity by allowing all nodes to authenticate a new node	User is authenticated once and access all the services posing a lot of security risks.
[32]	CoopEdge is a decentralised blockchain-based mechanism for edge computing. It ensures high network performance as the node with less latency is selected to mine the block.	No possible attacks were investigated, it only focuses on the performance such as CPU utilization. This make the level of security for this approach questionable.
[35]	BlockTDM is a blockchain-based platform trusted management system that make use of mutual authentication and multi-signature for encrypting sensitive data.	It involves complex problem solving which might result to high latency, affecting network performance.

Chapter 3

3.1 Introduction

This chapter explains the research methodology used to achieve the general aim and specific objectives outlined in chapter one. Section 3.2 presents the proposed Cloudlet-Blockchain Security Model overview. Section 3.3 discusses and highlights the role of the proposed agent layer as an intermediate layer between the edge device layer and cloudlet layer to ensure secure authentication between the two layers. Section 3.4 present the thesis proposed consensus mechanism, proof of trust, while section 3.5 explains how reputation of miners is controlled by taking away coins because of false validation. Finally, the summary of this chapter is provided in section 3.6.

3.2 Cloudlet-blockchain security model overview

This section presents an overview of the architectural representation of how the blockchain technology is incorporated into the cloudlet paradigm. The section further provide the details of each component and the role that each

component play in the Cloudlet-blockchain security model to ensure that the prescribed objectives are achieved.

The proposed agent layer between edge devices layer and cloudlet layer consists of an agent whose role is to authenticate the edge devices. This agent which plays a significant role in ensuring that only legitimate mobile users participate in communication over the cloudlet network, consists of a table with all the cloudlets ID as well as their existing number of coins. This table also contain the media access control (MAC) addresses for all the edge devices on the network. The cloudlet's IDs are used when generating password for message encryption between edge devices and cloudlets and the coins are used to determine reputation and experience for miners in the proposed blockchain technology

The primary purpose of the proposed blockchain technology is to ensure that both the stored data and data in transit are adequately secured on cloudlet. Each user data is stored in a sequence of blocks, with each block consisting of a previous block hash in the block header to inform or alert the blockchain. When data stored in any of the block is modified, the hash key changes and it can be easily determined if any node was compromised. This ensures that user's data stored in cloudlet is not easily modified. Also, if one of the nodes is compromised, it can easily get a non-modified chain from other nodes as each carries a copy of a chain. This proof of trust is adopted as the consensus mechanism for selecting a miner. In this consensus mechanism, being rich indicate good reputation and more experience. Therefore, not all nodes

participate in mining but the richest. However, to avoid a possible mistake that can be made by a rich entity, two miners are selected instead of one.

The rest of the sections in this chapter gives a detailed explanation how each process mentioned above in this section is achieved.

3.3 Authentication of nodes

Attacks such as denial-of-service (DOS) and distributed denial-of-service (DDOS) are caused by unauthorised users in a network. In the study, to ensure that entities participating the network are legitimate, they need to be registered and authenticated using registration server and an agent.

3.3.1 Registration server

A registration server(RS) is located in the cloud layer, as shown in figure 3.1. Its main role is to register mobile devices and cloudlet when they first join the network. It assigns them unique identifiers which are used by an agent for authentication before they can partake in any communication on the network.

Mobile device registration

Mobile users register themselves in the registration server(RS), located in the cloud as shown in figure 3.2. The registration process is detailed below:

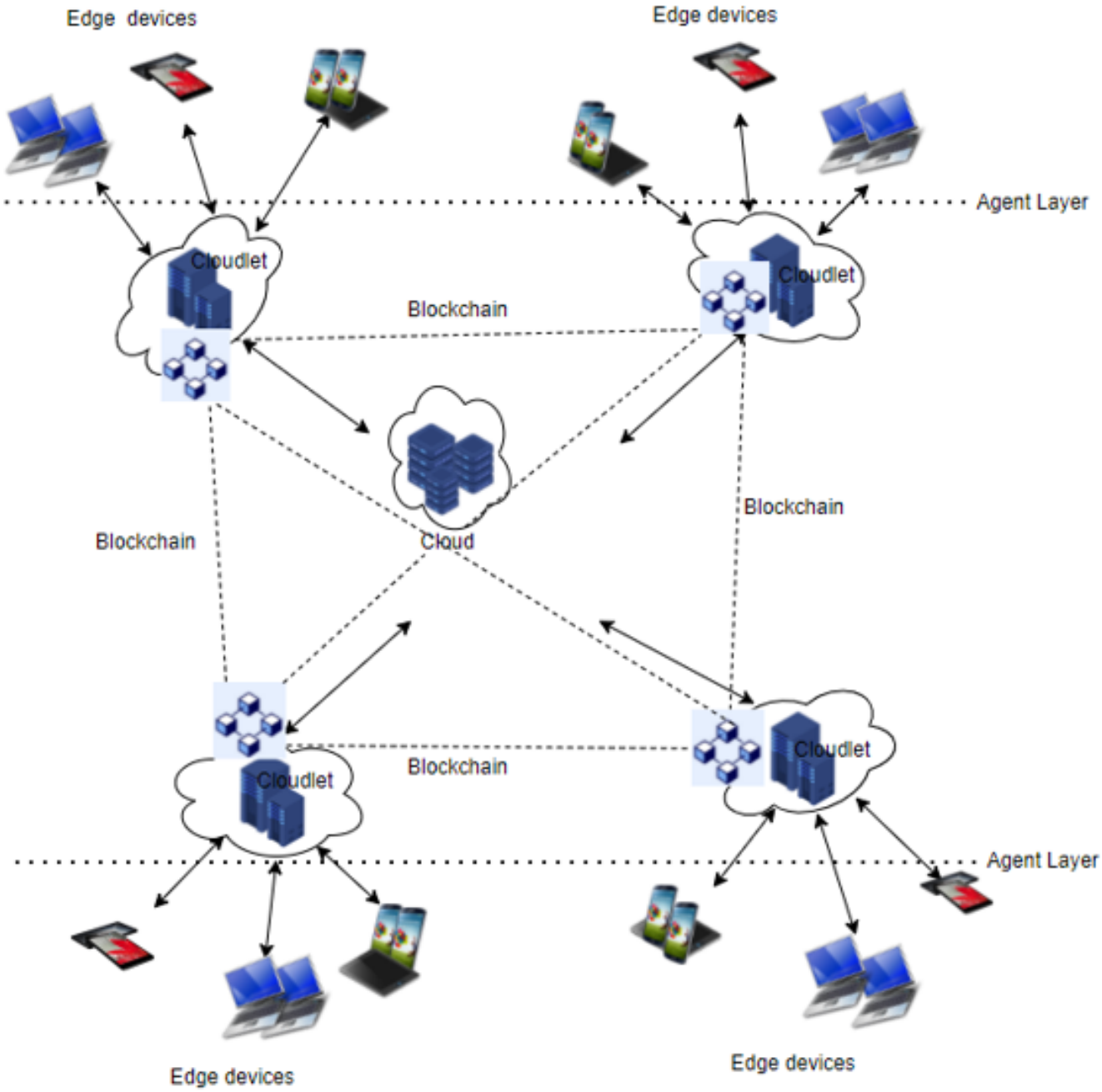


Figure 3.1: Cloudlet-Blockchain architecture

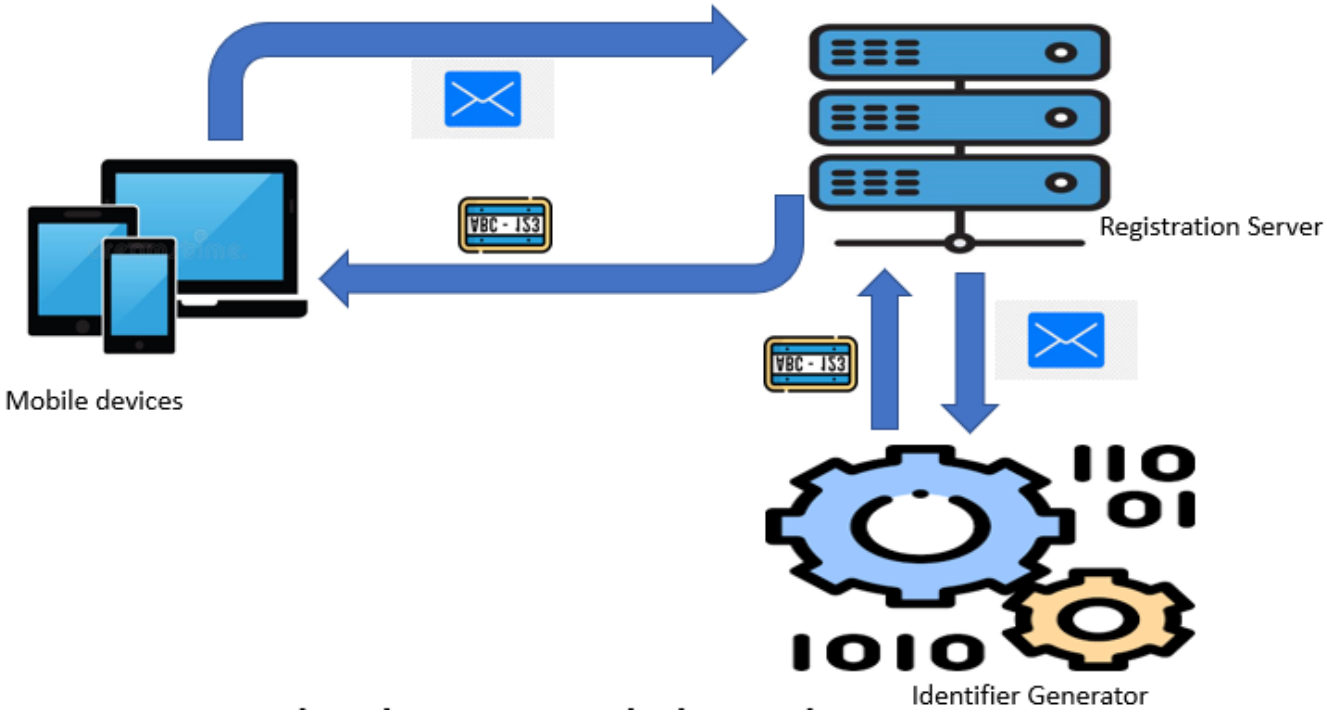


Figure 3.2: Mobile device registration

- Step 1: The mobile device initiates the registration process by sending a request to RS together with MAC address.
- Step 2: Upon receiving the request, the RS checks if the MAC address is not stored already and generates a unique ID for a mobile user, and send it securely.
- Step 3: A user stores the ID which will be used for further communication with the agent during the request to connect with the cloudlet.
- Step 4: If MAC address already exists, RS informs the mobile user, flag the MAC address for future reference in case it has been compromised. The communication is then terminated.

Cloudlet registration

Cloudlet needs to register in RS located on the cloud before joining the network as shown in 3.3. The steps on how registration take place are as follows:

- Step 1: The cloudlet sends a request to RS for registration.
- Step 2: The RS generate a unique ID and sends it to cloudlet securely.
- Step 3: The cloudlet stores this unique ID and will be used for validation by agent before any communication between it and mobile device is established.
- Step 4: The RS assigns the number of coins to each registered cloudlet but it is kept a secret to the cloudlet.

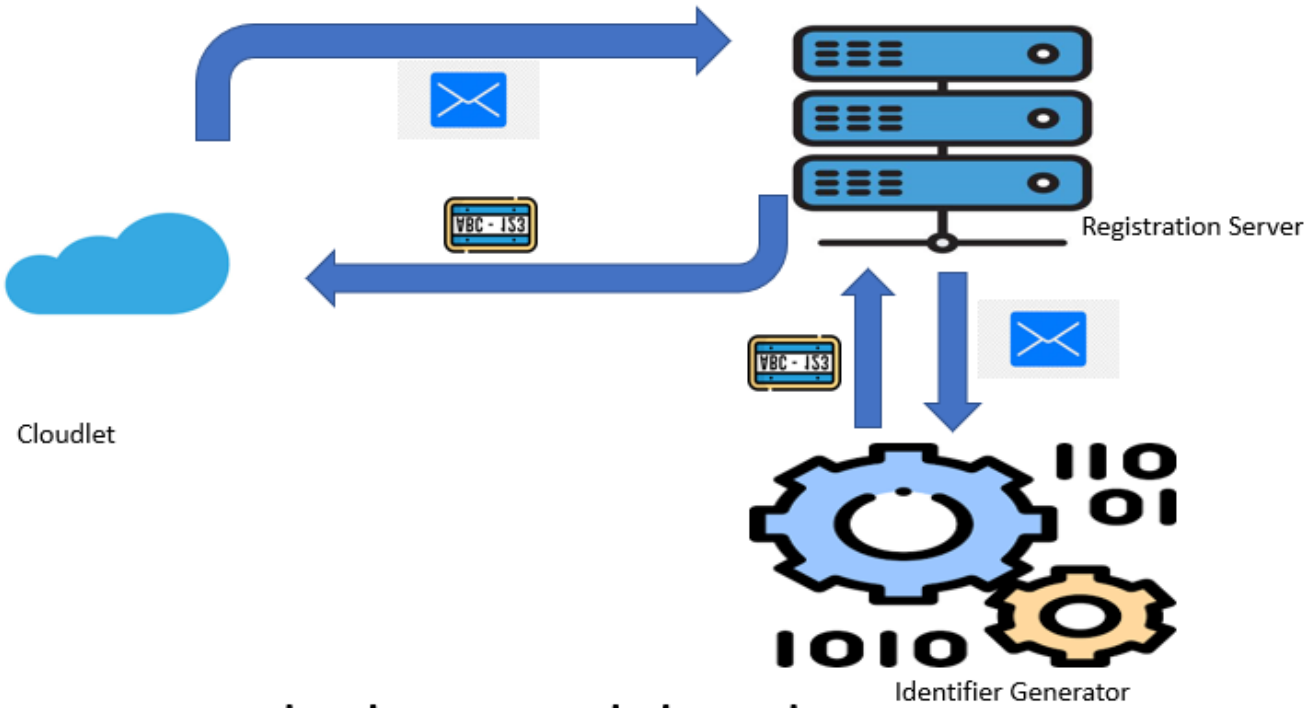


Figure 3.3: Cloudlet registration

The RS then sends a copy of the tables of the registered cloudlets and mobile user to the agent. These tables are then used to authenticate the mobile users and establish a connection between a cloudlet and a mobile user shown in figure 3.4.

3.3.2 Agent

Agent layer is a middle layer between mobile device layer and Cloudlet layer, as shown in figure 3.1. It consists of an agent that authenticate both cloudlet and mobile devices. An agent also plays a role in selecting a blockchain node to mine the block using coins stored in one of tables found in this layer.

Authenticating mobile devices

The agent layer authenticates the nodes before sending a message to cloudlets.

- Step 1: The mobile user request to connect to cloudlet from the agent.The MAC address is sent with the request.
- Step 2: The agent checks if such combination (ID and MAC address) exist.
- Step 3: If the mobile user has been validated, the agent check the cloudlets table to check two closer cloudlets with high number of coins. The importance of number of coins will be discussed later in this chapter.
- Step 4: The agent generates a password to be used by cloudlet and mobile user for communicating securely. The password is then encrypted

and send to mobile user and the two selected cloudlets.

- Step 5: Mobile user uses own ID to decrypt the password. The edge device can now attach the password to messages before sending them to the cloudlets, so they will use it to validate the sender.

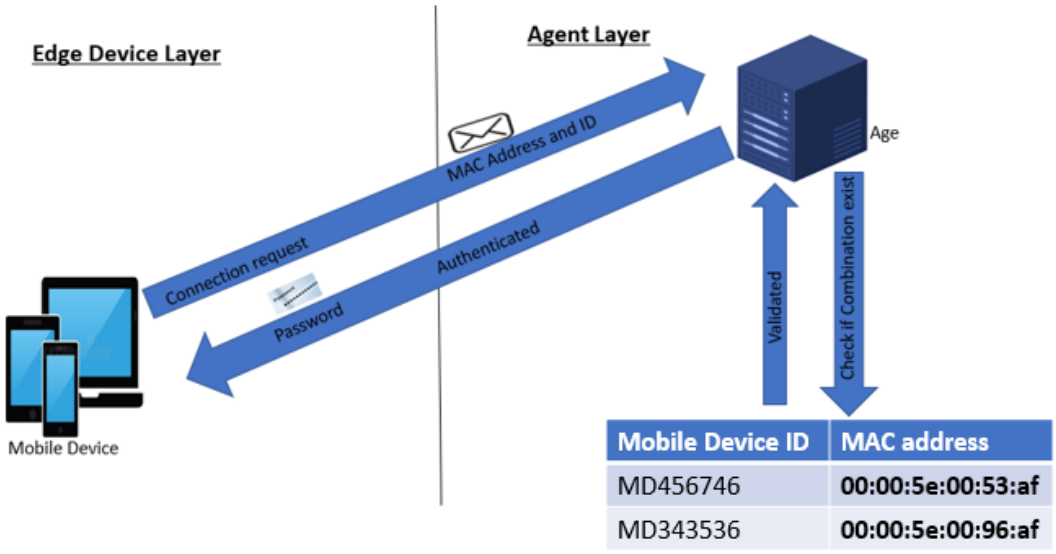


Figure 3.4: Mobile-Agent authentication

3.4 Proof of Trust

3.4.1 Proof of trust overview

Consensus is a process which allows every node in the blockchain network to agree upon connecting the new block to the chain. Proof of Stake(PoS) is a consensus mechanism that gives advantage of mining a block with less energy consumption [3]. In PoS, a richest node is given a right to mine a block. However, the richest node gets richer as it is constantly chosen due to the number of coins it acquire during mining. As a result, if this richest node is compromised, the whole network is affected without anyone noticing. Hence in our PoT, though the study adopts PoS concept of putting trust to richest nodes to mine the block, this is not done to a single node.

In this study, trust is defined as the ability of the entity to remain truthful in any given situation yielding non-bias, or questionable results. Trust is a measure of confidence on how much an entity will behave in an expected manner in a situation. Inconsistency in trust definition makes it difficult to establish a common explanation accommodating all the possible situation [40]. Two factors are to be used in determining trust - reputation and experience. For a node to be trusted, it should have good reputation and experience.

A node is said to be the richest if it holds a higher number of coins compared to the rest of the nodes. Because these coins are obtained at the end of successful mining, they are used to determine a node's experience and rep-

utation. The more a node is involved in mining a block, the more coins it will have as it gets awarded for every successful mining. This process speaks to node's experience. However, if a node fails message validation, it loses a number of coins. This ensures that though a node is experienced but it should also have a good reputation in terms of mining a block.

3.4.2 Miner selection

Cloudlet ID	Number of Coins
CLD1278937	19
CLD7879709	2

Figure 3.5: Mobile-Agent authentication

An agent from the agent layer is responsible for selecting the richest miners(cloudlets) to mine the block. The selected cloudlets will communicate with the authenticated mobile user, validate data given, mine the block and update the chain, and distribute the chain to all other node present on the network. The edge nodes will acquire coins based on their mining results with a possibility of them taken away due to false results. The following are steps on how the miner is selected:

- Step 1: An agent select two closets cloudlets with highest number of

coins from the table in the agent layer, an example of a table is shown in figure 3.5.

- Step 2: The agent encrypts the generated password to be used by the cloudlets and mobile users for secured communication, and send it to the 2 selected cloudlet.

- Step 3: The selected cloudlets uses their own unique IDs for decrypting the password. The cloudlets can now use the shared password to validate the mobile user and also to be validated by mobile user.

3.5 Block mining by cloudlets

3.5.1 Mobile user-cloudlets communication

This section gives a detailed description of the secured communication between cloudlets and mobile devices. As mentioned in the section above, the study uses elliptic curve signature to secure communication between the two entities.

Elliptic curve signature

Elliptic curve signature is a digital signature scheme that uses the elliptic-curve cryptography. It adopts the math of cyclic groups of elliptic curves over finite fields. The verify and sign algorithms which are part of this scheme relies on cryptographic elliptic curve point multiplication. In the verifying/sign process, key pairs (private and public) are used. Where a private key is used to sign the transaction and public key is used for verification. This private

key is generated using random integers in a range of [0, (n-1)], and public key is a point on the elliptic curve, calculated by multiplying cryptographic elliptic point by generator point.

In the study, elliptic curve digital signature is used to validate data from mobile user by cloudlet. Each user will have their own 2 key pairs (private and public). Signing and verification will occur as follows: mobile users will sign the message with their private key and send it to cloudlet. Cloudlet receives the message and decrypt it with mobile user's public key to verify it.

Message signing - mobile device

The description of how mobile device signs the message before sending it to cloudlets to prevent data modification. Figure 3.6 shows the process of signing a message by mobile devices which is detailed below:

- Both mobile device and cloudlet have public-private key pair. Private key is called signature key as it is used for signing process, and public key is called verifying key as it used for verification process.
- Step 1 : Mobile device input data to the hash function and generate hash value of the data.
- Step 2 : The hash value and the mobile device's private key are then fed to the signature algorithm which output digital signature.
- Step 3 : Both encrypted data and signature are sent to the selected cloudlets (blockchain nodes).

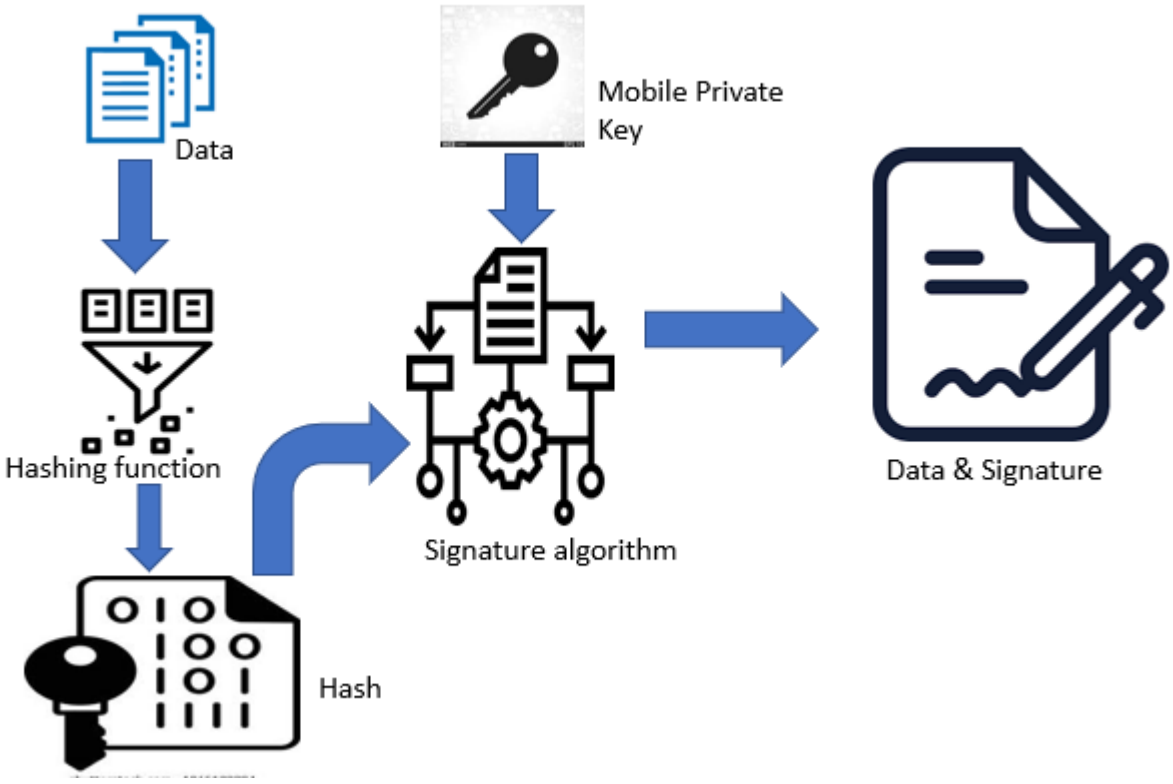


Figure 3.6: Message Signing process

Message verification - Cloudlets

Figure 3.7 shows the process of verifying a message by cloudlets which is detailed below:

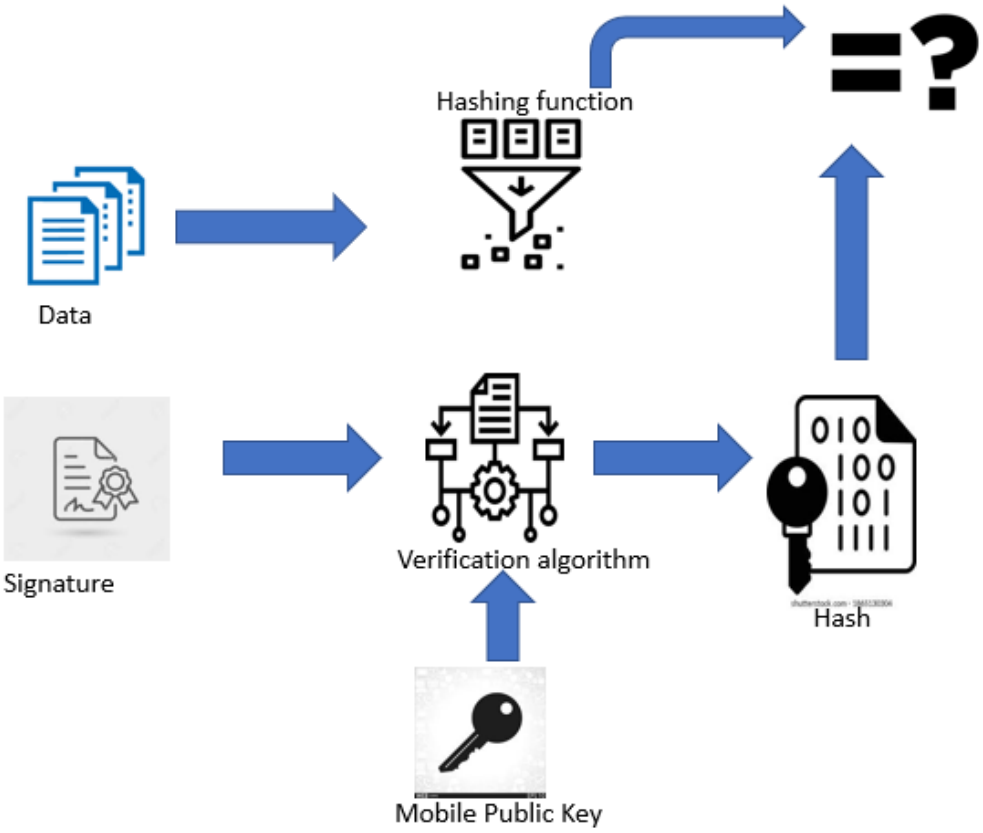


Figure 3.7: Message verification process

- Cloudlets input digital signature received from mobile device, and mobile device's verification key to the verification algorithm.
- Cloudlet also runs the same hash function on received data and generate hash value.

- The hash values and output from the verification algorithm are compared to determine if the mobile device that sent the message is legitimate.

3.5.2 Verification agreement

Impersonation attack is one of the phishing security threat where malicious users poses as known node. To prevent effects of such attacks, it is assumed that when one of the nodes is compromised, this results to selected richest miners disagreeing on message validation results. This sections provide details on what happens when the two cloudlet does not reach consensus when validating a message from the mobile device user. The situation is handled by following the steps:

- Step 1 : The agent from the agent layer is notified that there has been some disagreement between the two cloudlets.
- Step 2: The agent selects a third cloudlets with the highest number of coins, that is apart from the initially selected cloudlets.
- Step 3: The message from mobile device user is sent to the third selected cloudlet for verification.
- Step 4 : After the third cloudlet has verified, the one in disagreement loses all the coins to maintain reputation and experience factors.
- Step 5: The cloudlets that properly validated gets awarded with coins.
- Step 6: Number of coins are updated on the coin table in the Agent layer.

3.5.3 Adding a block to a chain

This subsection explains how mobile user's data is added to blockchain after it has been validated how the blockchain is then distributed.

The addition process begins with the cloudlet having the highest number of coins, previously selected in section 3.4, creating a block using mobile user's data by attaching a timestamp and the hash key of the previous block to form a chain. The chain is then distributed to all other nodes (cloudlets). This ensures that each node has an updated copy of blockchain. Furthermore, if any of the data stored on the nodes has been compromised, then the security issue is mitigated.

3.6 Summary

This chapter discussed how blockchain is used to secure stored data in cloudlet. This new approach makes use of a consensus mechanism based on reputation and experience which is decided by the number of coins each node(cloudlet) holds. It also adopts additional layer called agent layer, located between mobile device layer and cloudlet layer, which consist of an agent responsible for authenticating mobile devices and selecting a miner based on consensus mechanism.

The next chapter presents and discusses results obtained after applying the proposed blockchain technology to a cloudlet network to ensure enhanced security.

Chapter 4

4.1 Simulation environment

4.1.1 EdgeCloudSim architecture

In this study, the EdgeCloudSim simulator was adopted to simulate the entire methodological system processes discussed previously in chapter 3 of this study. EdgeCloudSim was introduced by Cagatay Sonmez as a simulator that support different functionalities such as network modelling - LAN device mobility model and WLAN load generator model [41].

EdgeCloudSim offers a modular architecture where each module focuses on a specific part of edge computer with clearly defined interfaces to other modules. This simulator has five modules as listed below and also shown in figure 4.1:

- Core simulation module - This module's main task is loading and running Edge computing scenarios from the configuration files. It also saves the results into a CSV file.

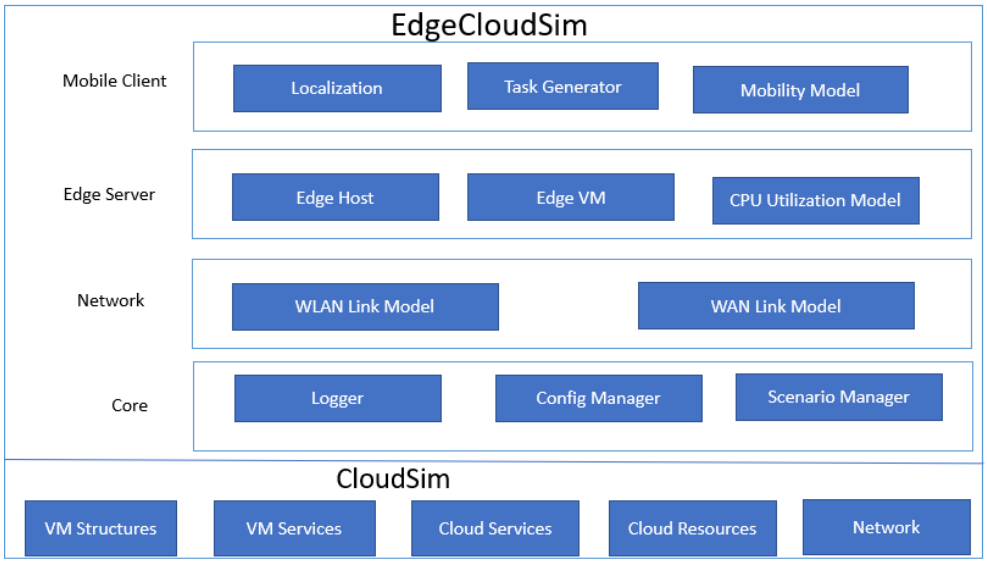


Figure 4.1: Edgecloudsim block diagram

- Networking module - This module is responsible for handling the transmission of data in the WAN and WLAN. It takes care of transmission delay during uploading and downloading of that between Mobile devices and Cloudlet, as well as between Cloudlet and Cloud.

- Edge orchestrator Module - This is decision maker of the system and it has a strong relationship with edge server layer. It handles client's request using all the information collected from other modules.

- Mobility module - This module is responsible for updating mobile devices location. Each mobile device has X and Y coordinates which are update in a managed hash table. It also records locations where Wi-Fi access points are utilized.

- Load generator module- This module is responsible for task genera-

tion using provided configurations. The mobility and load generator modules are the main components which provide input to other components.

Modular design and open source code base of EdgeCloudSim allow us to incorporate the specific needs in our simulator experiments.

Edgecloudsim class hierarchy

The study made use of the following classes, and then modified them to meet our architectural design goal, while we leave the rest of the classes the same as they are in the work of [2]. BasicEdgeOrchastractor extends the abstract class, EdgeOrchastractor. It implements basic algorithm which are first/next/worst/ random fit algorithms to select a suitable Virtual Machine (VM) for task offloading. The class was modified such that virtual machine (VM) is selected base on a number of coins it holds. The DefaultMobileDeviceManager extends MobileDeviceManager Class. Its main function is to submit tasks to appropriate device (cloudlet/cloud). It is also responsible for taking needed actions after processing tasks. This is where a digital signed message is attached to a task before sending the task to cloudlets. Figure 4.2 shows a class hierarchy for EdgeCloudSim, as well as highlighted classes that were modified to implement our work.

Simulation configurations

Due to many parameters used in the original EdgeCloudSim simulator, managing these parameters programmatically will be difficult, hence configura-

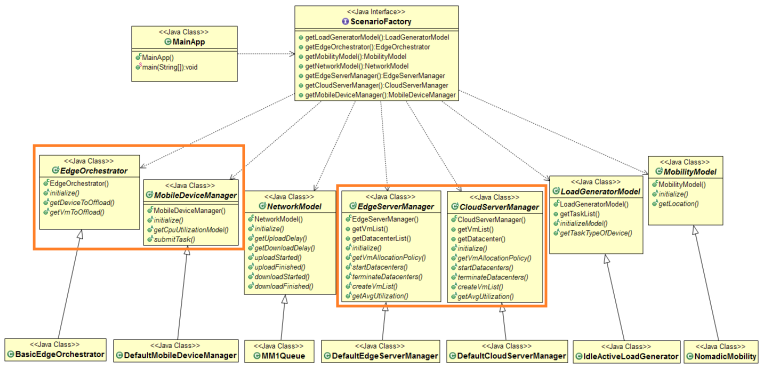


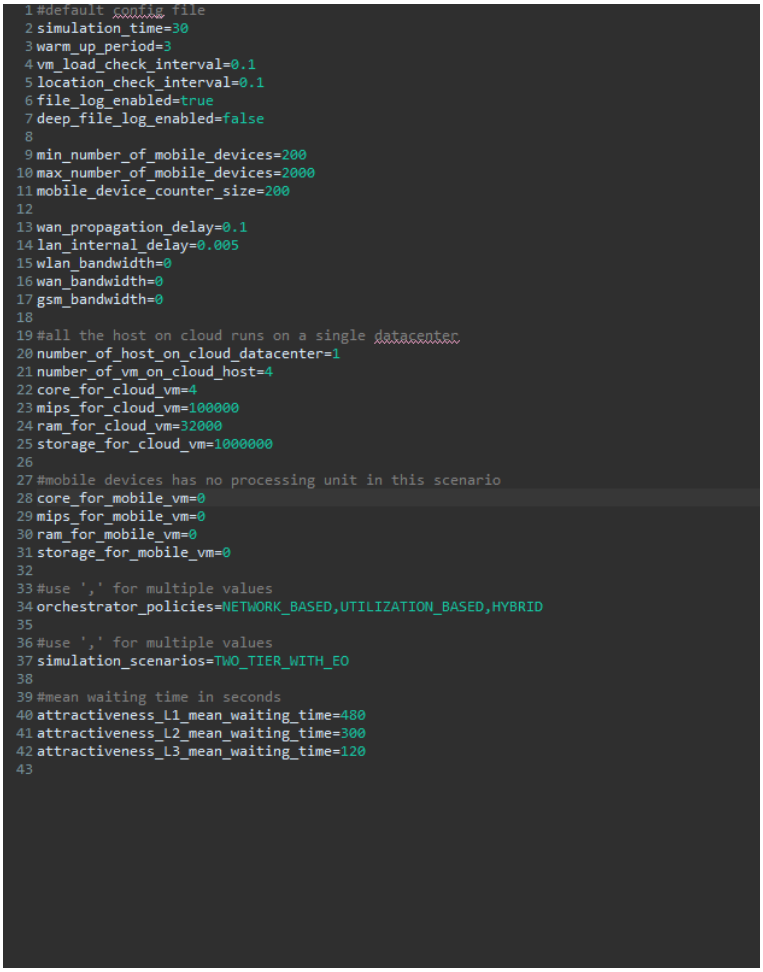
Figure 4.2: Edgecloudsim class hierarchy [2]

tion files are used to manage the parameters. EdgeCloudSim reads parameters dynamically from the following files:

- config.properties: Simulation settings are managed in configuration file. Figure 4.3 shows details of what is contained inside this file.

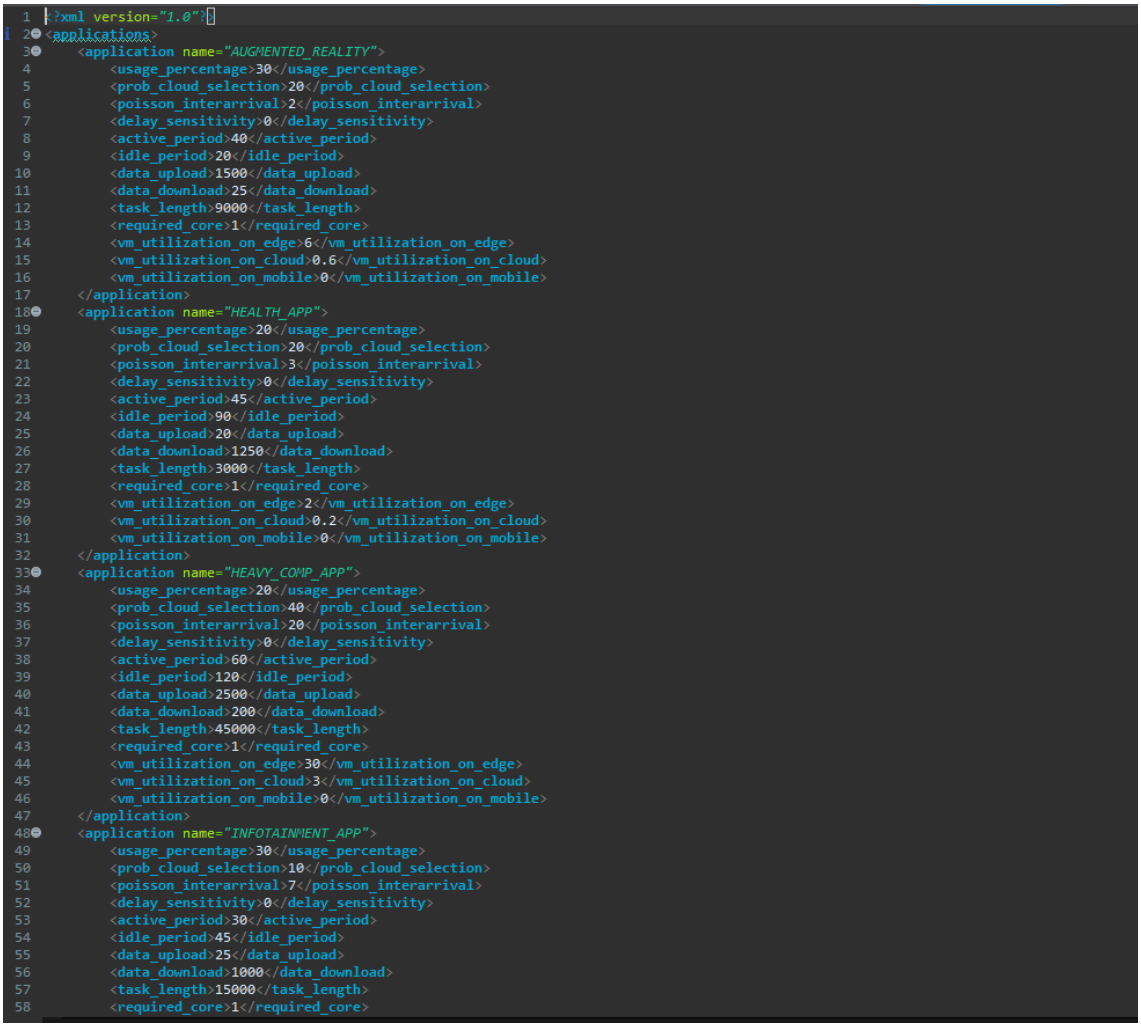
- applications.xml: Application properties are stored in xml file. To evaluate the performance of the network as well as the impact of the proposed security model, applications, namely Augment Reality,Healthy, Heavy Comp, and Infotainment, were used which are provided by the simulator. Figure 4.4 shows XML with detailed applications' specifications. Idle/active task generation pattern is used to mimic a real life scenario where mobile devices does not generate request to service continuously. A task is generated during active time. A generated task does not have fixed length, it is based on the file size that is being downloaded or uploaded.

- edgedevices.xml: Edge devices (datacenters, hosts, VMs etc.) are defined in xml file. Figure 4.5 show details of what is contained inside



```
#default config file
simulation_time=30
warm_up_period=3
vm_load_check_interval=0.1
location_check_interval=0.1
file_log_enabled=true
deep_file_log_enabled=false

min_number_of_mobile_devices=200
max_number_of_mobile_devices=2000
mobile_device_counter_size=200

wan_propagation_delay=0.1
lan_internal_delay=0.005
wlan_bandwidth=0
wan_bandwidth=0
gsm_bandwidth=0

#all the host on cloud runs on a single datacenter
number_of_host_on_cloud_datacenter=1
number_of_vm_on_cloud_host=4
core_for_cloud_vm=4
mips_for_cloud_vm=100000
ram_for_cloud_vm=32000
storage_for_cloud_vm=1000000

#mobile devices has no processing unit in this scenario
core_for_mobile_vm=0
mips_for_mobile_vm=0
ram_for_mobile_vm=0
storage_for_mobile_vm=0

#use ',' for multiple values
orchestrator_policies=NETWORK_BASED,UTILIZATION_BASED,HYBRID

#use ',' for multiple values
simulation_scenarios=TWO_TIER_WITH_EO

#mean waiting time in seconds
attractiveness_L1_mean_waiting_time=480
attractiveness_L2_mean_waiting_time=300
attractiveness_L3_mean_waiting_time=120

```

Figure 4.3: Simulation settings



```xml
<?xml version="1.0"?>
<applications>
    <application name="AUGMENTED_REALITY">
        <usage_percentage>30</usage_percentage>
        <prob_cloud_selection>20</prob_cloud_selection>
        <poisson_interarrival>2</poisson_interarrival>
        <delay_sensitivity>0</delay_sensitivity>
        <active_period>40</active_period>
        <idle_period>20</idle_period>
        <data_upload>1500</data_upload>
        <data_download>25</data_download>
        <task_length>9000</task_length>
        <required_core>1</required_core>
        <vm_utilization_on_edge>6</vm_utilization_on_edge>
        <vm_utilization_on_cloud>0.6</vm_utilization_on_cloud>
        <vm_utilization_on_mobile>0</vm_utilization_on_mobile>
    </application>
    <application name="HEALTH_APP">
        <usage_percentage>20</usage_percentage>
        <prob_cloud_selection>20</prob_cloud_selection>
        <poisson_interarrival>3</poisson_interarrival>
        <delay_sensitivity>0</delay_sensitivity>
        <active_period>45</active_period>
        <idle_period>90</idle_period>
        <data_upload>20</data_upload>
        <data_download>1250</data_download>
        <task_length>3000</task_length>
        <required_core>1</required_core>
        <vm_utilization_on_edge>2</vm_utilization_on_edge>
        <vm_utilization_on_cloud>0.2</vm_utilization_on_cloud>
        <vm_utilization_on_mobile>0</vm_utilization_on_mobile>
    </application>
    <application name="HEAVY_COMP_APP">
        <usage_percentage>20</usage_percentage>
        <prob_cloud_selection>40</prob_cloud_selection>
        <poisson_interarrival>20</poisson_interarrival>
        <delay_sensitivity>0</delay_sensitivity>
        <active_period>60</active_period>
        <idle_period>120</idle_period>
        <data_upload>2500</data_upload>
        <data_download>200</data_download>
        <task_length>45000</task_length>
        <required_core>1</required_core>
        <vm_utilization_on_edge>30</vm_utilization_on_edge>
        <vm_utilization_on_cloud>3</vm_utilization_on_cloud>
        <vm_utilization_on_mobile>0</vm_utilization_on_mobile>
    </application>
    <application name="INFOTAINMENT_APP">
        <usage_percentage>30</usage_percentage>
        <prob_cloud_selection>10</prob_cloud_selection>
        <poisson_interarrival>7</poisson_interarrival>
        <delay_sensitivity>0</delay_sensitivity>
        <active_period>30</active_period>
        <idle_period>45</idle_period>
        <data_upload>25</data_upload>
        <data_download>1000</data_download>
        <task_length>15000</task_length>
        <required_core>1</required_core>
```

Figure 4.4: Configurations for applications

this file. Each edge server has one host operating 2 VMs with 10 Giga instructions per seconds CPU powers.

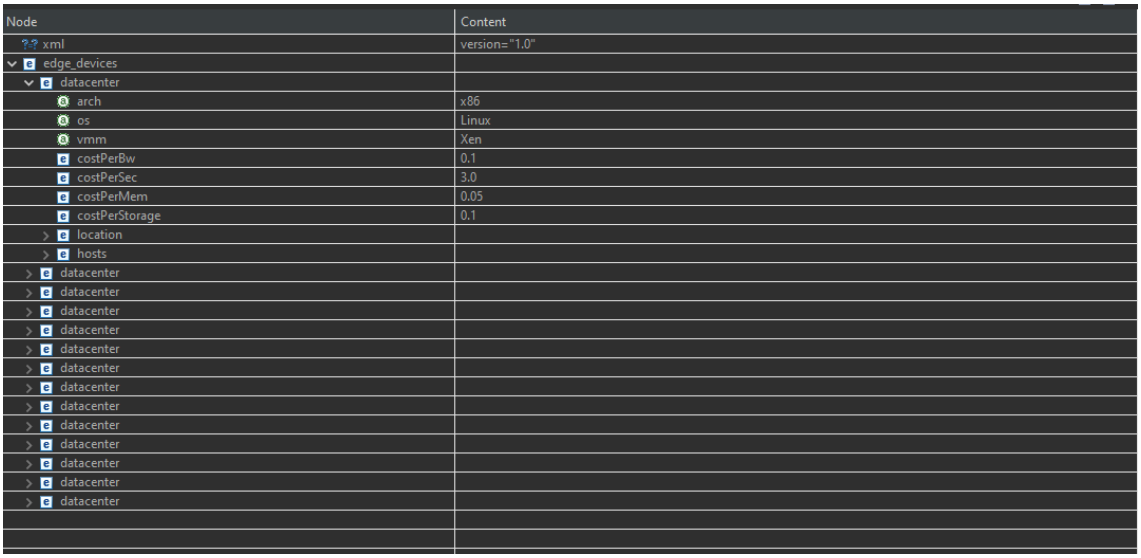


Figure 4.5: Configurations of Edge devices

Simulation parameter configurations

The virtual environment adopted from [2] is similar to a university campus with students walking around and making request to buildings with edge servers and with each building having a wireless access point. The study adopted a two-tier with Edge Orchestrator(EO), where mobile devices can offload to other edge servers located in different buildings. The number of devices represents mobile device users sending request to the edge servers, either downloading or uploading files. The task generated does not have a fixed length, it is based on file size being downloaded or uploaded. Idle/active task generation pattern is used to mimic real life scenario where mobile de-

vices does not generate request to cloud service continuously. A task is only generated during active time. Table 4.1 lists simulation parameters used to simulate the work.

Table 4.1: Simulation parameters.

Parameters	**Values**
Simulation time	30 minute
WAN/WLAN Bandwidth(Mbps)	20/300
Number of repetitions	10
Number of places	3
Active/Idle period of the user (second)	45/15
Provisioning algorithm on edge	Least-loaded and Number of Coins
Probability of cloud offloading	0.1
Number of VMs per edge server	8
CPU speed per edge/ cloud VM	10/200 GIPS
Average Data Size for Upload/ Download (KB)	1500/15

4.2 Proposed security framework implementation

This section provides details of how the proposed security framework is implemented using EdgecloudSim simulator. A computer with 8GB of RAM with Eclipse version 4.25 was used. Classes that comes with the sample application 3 of the simulator were modified.

4.2.1 Cloudlet selection

Cloudlet selection is a process of selecting a miner(cloudlet) that will partake in verifying a message sent from mobile devices to cloudlets. This selection is based on the stake (number of coins) each node holds. The detailed steps of this process is discussed in section 3.4.2. In this subsection, a detailed explanation on how this process is implemented is given. The process starts by implementing the Agent layer with an agent responsible for carrying out the cloudlet selection process, and how the agent selects miners based on the stake it has. The study implemented the Agent layer with an Agent as shown in figure 4.6. All these methods shown in this figure are implemented at the Edge methods as a layer between mobile device layer and Cloudlet layer. As mentioned in the previous chapter, Agent uses number of coins to select miners, we start by defining a method called GenerateCoins on the DefaultEdgeServerManager class as shown in figure 4.7. To imitate a real life scenario, coins were randomly assigned to cloudlets. This method was then invoked to assign the coins to cloudlet in the CreateVMList method, part of the same class. This is shown in figure 4.8.

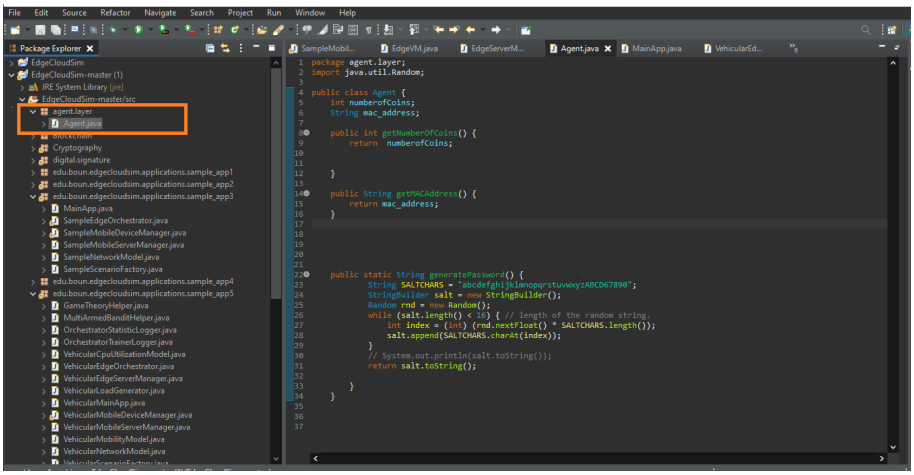


Figure 4.6: Agent layer

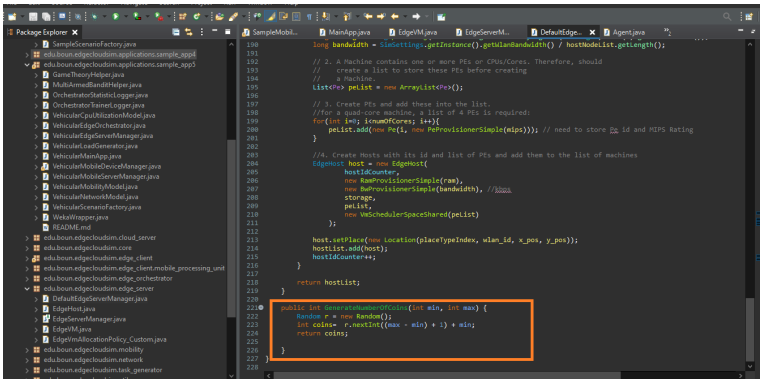


Figure 4.7: Method for generating number of coins

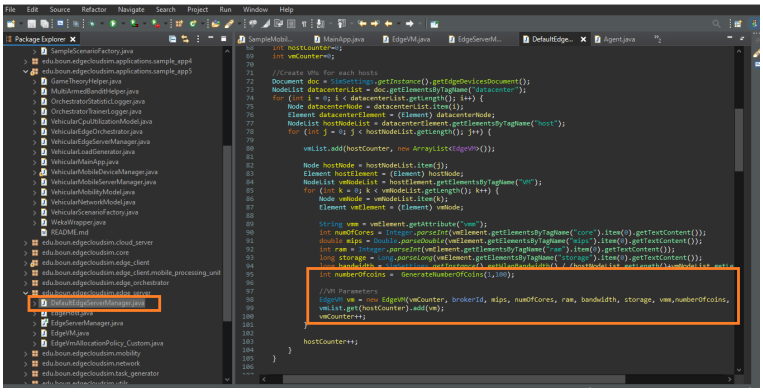


Figure 4.8: Generation of coins

Figure 4.9 shows how the agent uses coins to select the richest nodes. To achieve this functionality, the researcher modified the GetVMToOffload method from SampleEdgeOrchastrator class which is subclass of EdgeOrchestrator super class.

Figure 4.9: VM selection based on number of coins

4.2.2 Mobile device to cloudlet communication

As discussed in Section 3.5.1, in the proposed security model, mobile devices should digitally sign a message before sending it to cloudlet. To confirm that

the data was not modified during transit, a cloudlet needs to confirm the message validity by verifying the signature. This subsection explains how this process is implemented in the simulated environment. The researcher implemented five classes responsible for signing message by mobile device users and verified by cloudlet as shown in figure 4.10. The mobile device

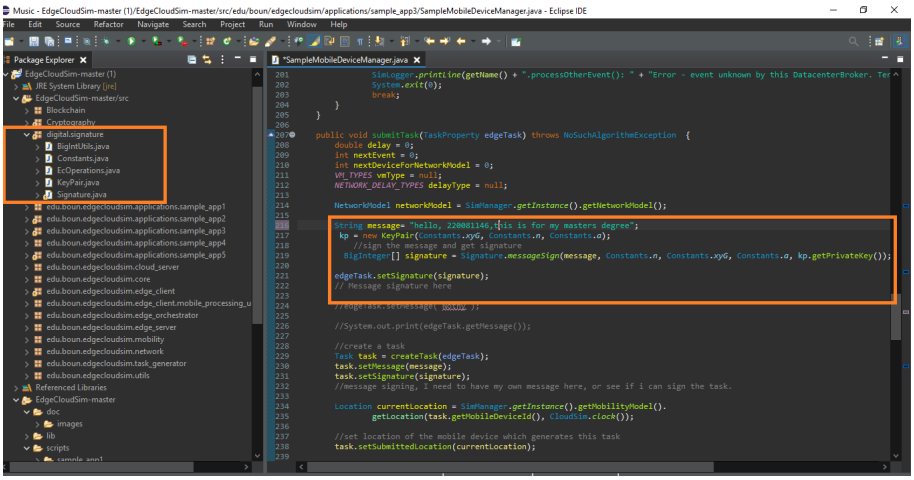


Figure 4.10: Digital signature classes

needs to digitally sign the message before sending it to cloudlet. To implement this function, that is the digital signing of the message, we modified the methods as shown figure 4.10.

Upon receiving a message from mobile device by Cloudlets, a message signature needs to be verified to ensure there was no data modification. To achieve this functionality, the researcher invoked verification method as shown in 4.12. Two cloudlets take part in verifying the message and if they do not reach consensus, a third cloudlet is selected. If two cloudlets reach consensus, the third one loses the number of coins. The method was implemented to

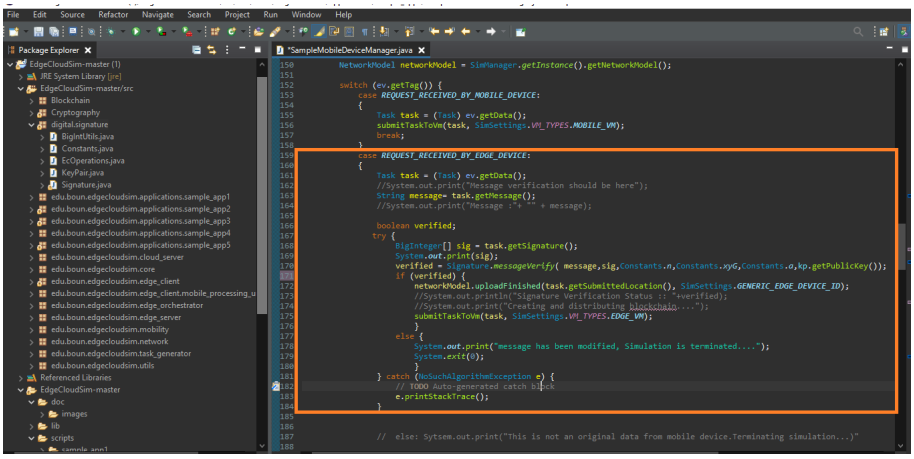


Figure 4.11: Digital signature verification

assign the number of Coins from nodes that successfully verified and take away coins to a node that unsuccessful verified. This allows the researcher to control the level of trust and experience on the nodes.

4.2.3 Blockchain

Protecting data in transit and not when stored does not guarantee that the network is secured. Hence after the message is verified it is then stored in blockchain to ensure that it is not susceptible to data modification as discussed in Section 3.5.3. This subsection explains how we implemented a functionality of storing data from a mobile device to a blockchain. Figure 4.13 shows all classes that are implemented at cloudlet layer level.

Once the cloudlet has verified that the message has not been altered as shown in the previous subsection, it creates a block and add it to a chain. This is demonstrated in figure 4.14. Each block holds a hash key of the previous block and this creates a blockchain.

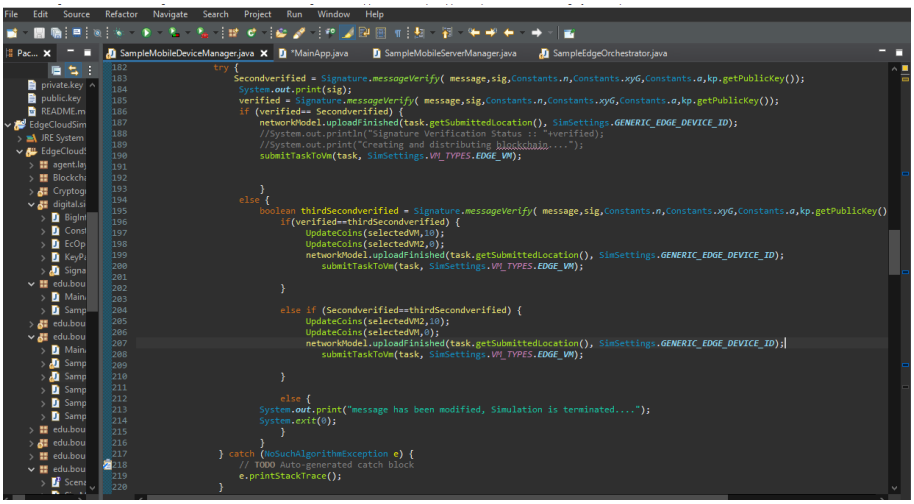


Figure 4.12: Digital signature verification

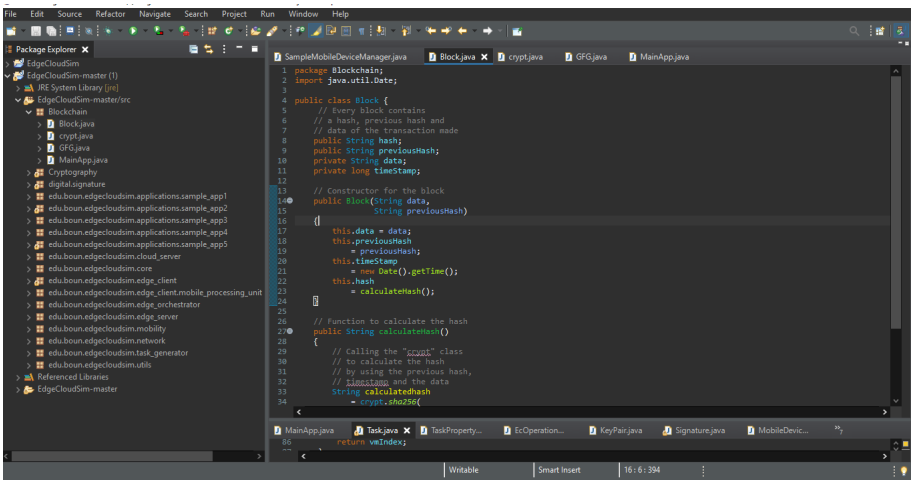


Figure 4.13: Digital signature verification

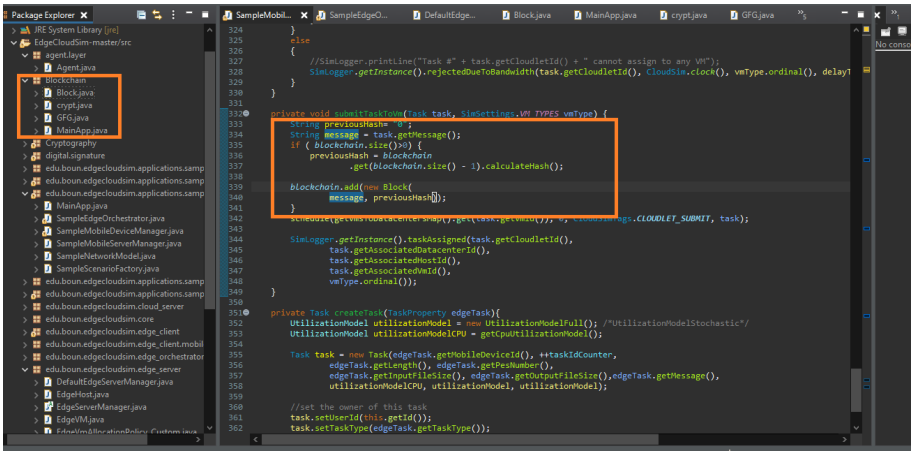


Figure 4.14: Block creation

Chain Validation

After a block is added on the chain, the blockchain is distributed to all other nodes (cloudlets) on the network. However,there is a need to confirm that a legitimate chain is distributed. Hence, the researcher implemented a process of verifying the validity of the chain. Figure 4.15 shows a method that was used to check for data modification on the chain.This method was used after adding the block to the chain as shown in figure 4.16

4.3 Simulation results and discussion

In the study, result analysis evaluate the aspect of security model which addresses the following questions: how secure the network is with the proposed model implemented and the performance of the network upon implementing proposed security model. To test the security model in the simulated network environment, the researcher implemented a method that act as a

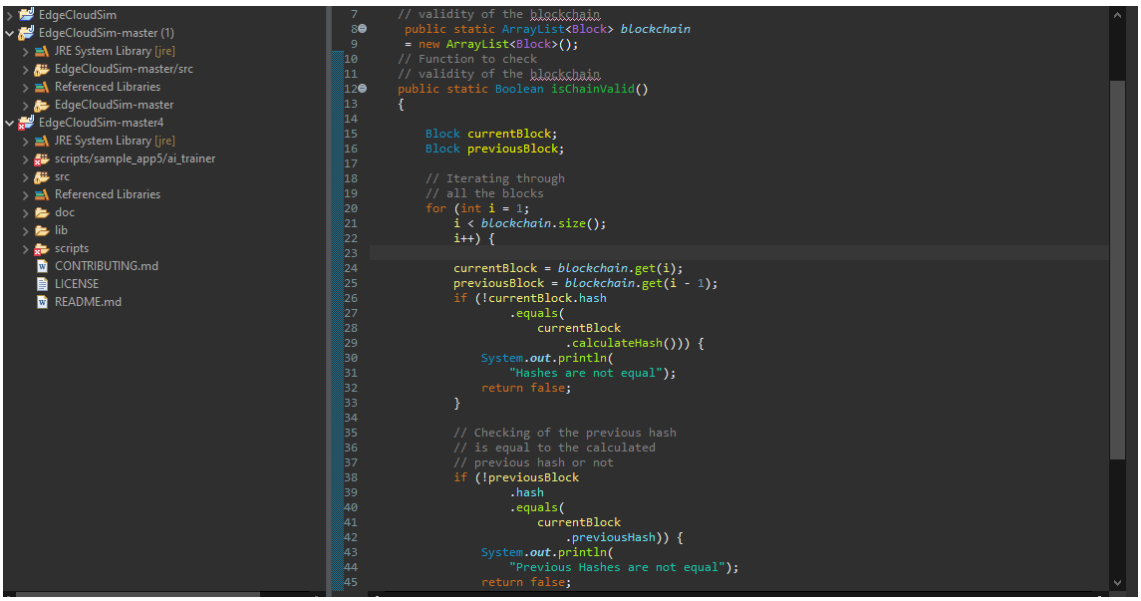


Figure 4.15: Method for verifying a chain

```
private void submitTaskToVm(Task task, SimSettings.VM_TYPES vmType) {
    String previousHash= "0";
    String message = task.getMessage();
    if ( blockchain.size()>0) {
        previousHash = blockchain
                .get(blockchain.size() - 1).calculateHash();

    blockchain.add(new Block(
            message, previousHash));
    }

    ChainValidation chainValidation = new ChainValidation();
    boolean chain =chainValidation.isChainValid();
    if(chain) {
    schedule(getVmsToDatacentersMap().get(task.getVmId()), 0, CloudSimTags.CLOUDLET_SUBMIT, task);

    SimLogger.getInstance().taskAssigned(task.getCloudletId(),
            task.getAssociatedDatacenterId(),
            task.getAssociatedHostId(),
            task.getAssociatedVmId(),
            vmType.ordinal());
    }
}
```

Figure 4.16: Chain validation method invocation

middle man. This method modifies the data while in transit, that is before it reaches the cloudlet. Figure 4.17 show a method that was used to generate messages for every mobile device. This method reads a file and assign a message to each task associated with task sent to cloudlet by the mobile users. Figure 4.18 shows a method that was used to modify a message while

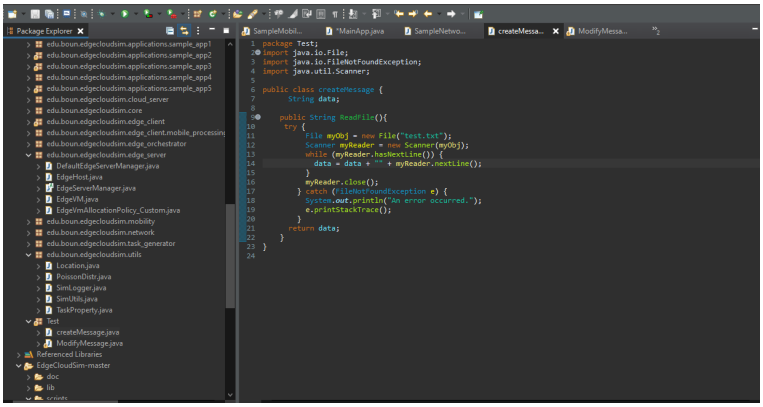


Figure 4.17: Method for creating a message

in transit before it reaches cloudlets.

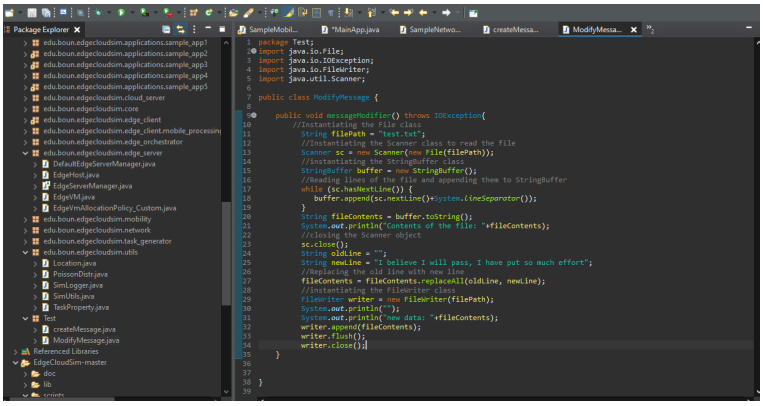


Figure 4.18: Method for message modification

4.3.1 Security analysis

The aim of the work is to develop a security model to secure data in transit and when stored in cloudlet. To test the proposed security framework solution, the following scenarios were implemented:

- Scenario 1: The proposed security model ensure integrity for data in transit.

  Proof: Data from mobile devices should be offloaded to cloudlets without modification. Let's assume that the intruder impersonates the mobile device and sends the request to cloudlet. For digital signature, public and private keys from Elliptic curve scheme were used as discussed in Section 3.5.1. The cloudlet will verify the digital signature of the message, using mobile device's public key. Since the unauthorised user does not have access to mobile device's private key, hence the attempt will not succeed.

- Scenario 2: The security model ensures that trust is not restricted to just one node in case it is compromised.

  Proof: Let's assume an intruder fakes the number of coins it holds to stand more chances of being selected as a miner. During block mining, the fake node will verify the signature and approve the message even though it has been modified. The second miner which is legitimate will reject the message since it has been modified. This results to disagreement between the two nodes, legitimate and fake nodes. As discussed in section(3.5.2), if two miners do not reach consensus a third node is selected to resolve to the conflict by completing the whole process of

signature verification on a message. Because the message is not the original message from mobile device, then the third node will reject the message. Since number coins in a node determine the level of trust experience, the fake node will lose all its coins due to false verification and never stand any chance to participate in the mining process again.

- Scenario 3 : The security model ensure integrity in stored data.
  Proof: Let's assume that a node (cloudlet) is compromised and an intruder manages to modify data stored. Since data is stored in blockchain, if data in one block is modified, it hashes key changes and this breaks the chain. With this in context, it can then be determined that data in storage was modified, and node is compromised.

4.3.2 Performance evaluation

In this section, the evaluation of the applications based on the three-performance metrics was done. For each metric, a graph was provided that corresponds to an application and the average value for all the values of the current performance metric. Figure 4.19 show the results generated from the simulated network with proposed security model which were stored on a text file.

It is important at this point to define a better performing blockchain mechanism.Less complex blockchain consensus mechanism has proven a reduced latency [42]. Edge servers provide better performance to end-users with decreased latency but that cannot be said for the cloud computing; cloudlet characteristics is offering cloud services with less time and improved throughput[43].

This means that better results should be produced for service time, processing time as well as Network latency with the blockchain mechanism implemented.

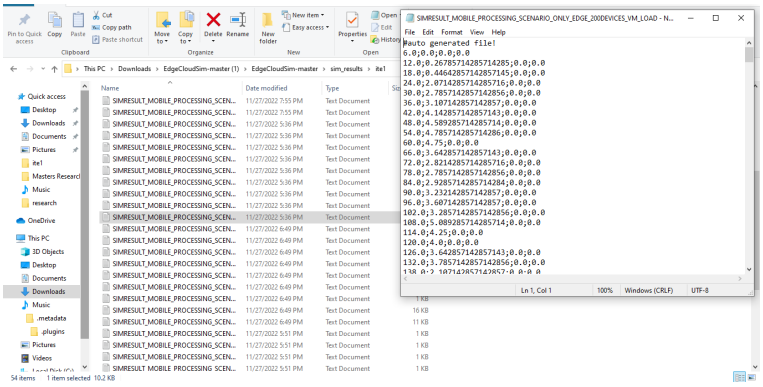


Figure 4.19: Results in CSV format

Processing time

The researcher began by testing how the network behave in terms of processing time and when the response sensitive applications are offloaded and executed at the cloudlet level. The main interest is at cloudlet level because this is where the research implemented the main feature of the security model. Figure 4.20 to figure 4.23 present results for processing time relative to the different applications. These results were obtained when the four applications were offloaded. The processing time for mobile device is zero throughout, this is because all the tasks were offloaded to cloudlet. This can be observed from all these graphs, the higher the number of devices the higher the processing time, this is because there are more task to be processed. It is important to note that these results shown are for the overall processing time of tasks offloaded to cloudlets, it does not present processing time of task offloaded

by number of devices per cloudlet. It was observed that with higher number of task offloaded, there are more instances where nodes does not reach consensus and raises a need to introduce a third node as per our model discussed in section 3.5.2. This gives a strong belief that more nodes participating in mining, results in higher processing time. Since the higher processing time results in more energy consumed. Hence, the model of using a limited number of nodes in verification performs better as compared to proof of work as far as energy consumption is concerned.

The cloudlet was originally invented to overcome the limit of cloud computing which was introduced to overcome the limitations of mobile device. In figure 4.24, the average processing time for all the applications is lower compared to the other two scenarios where only mobile - tasks are executed on a mobile device, and hybrid - task is executed in either edge or central cloud. This is expected as the cloudlet distributes the load across all the cloudlets. These results shows that the proposed solution has no negative impact on the performance of the network as far as the processing time is concerned. It still performs as expected which is to have lesser processing time irrespective of the number of mobile devices.

Network delay

The delays, here, simply means the time for which the processing of a particular user data takes place. In figure 4.25 the results based on average WLAN delay is shown. Due to the way the setup is designed, it is expected for mobile devices to be zero as they do not transmit any data over the net-

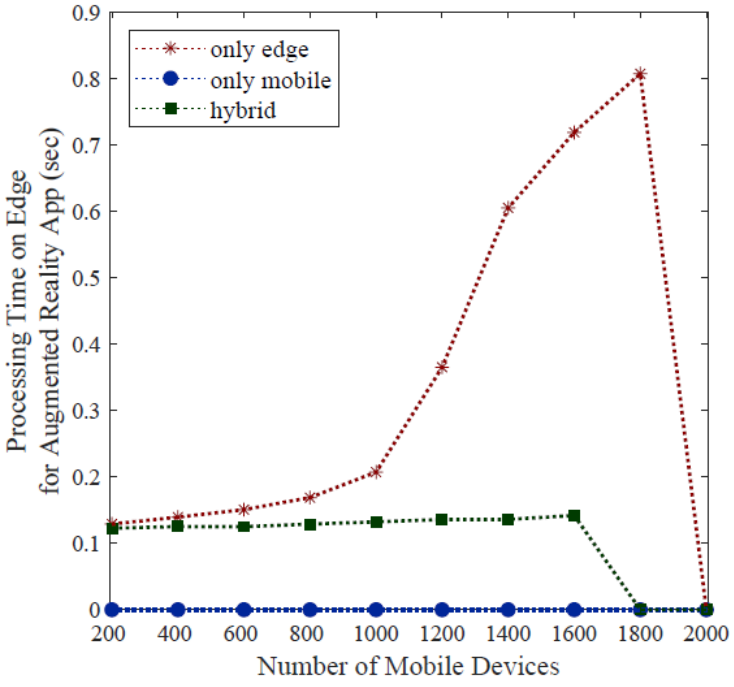


Figure 4.20: Processing time for Augmented Reality App results.

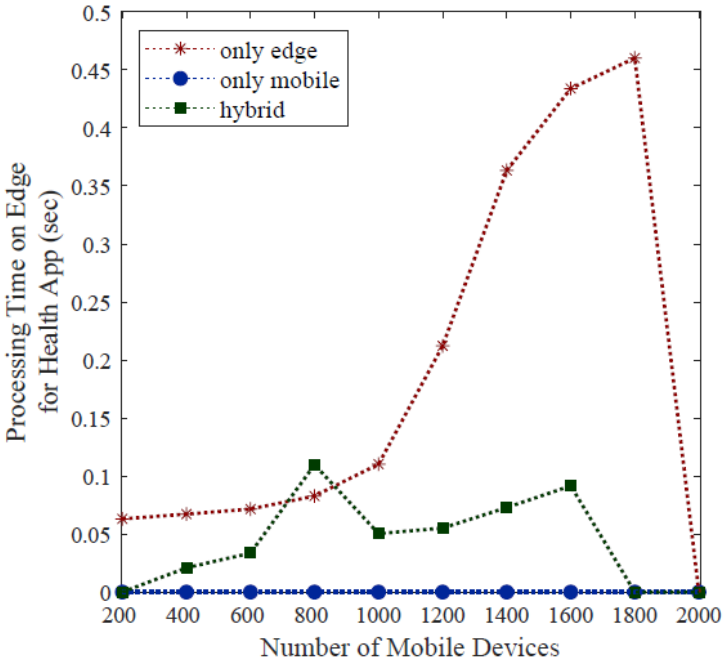


Figure 4.21: Processing time for Health App results

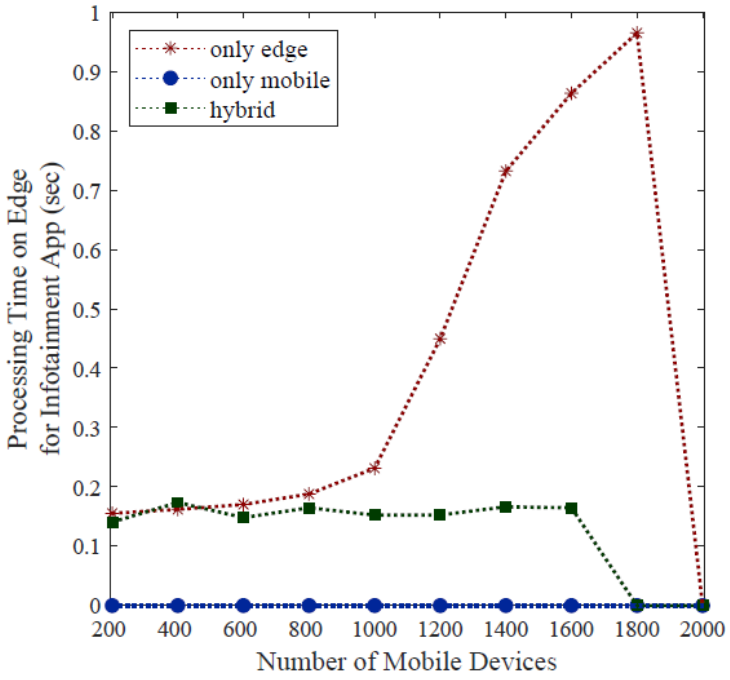


Figure 4.22: Processing time for Infoteinment App results

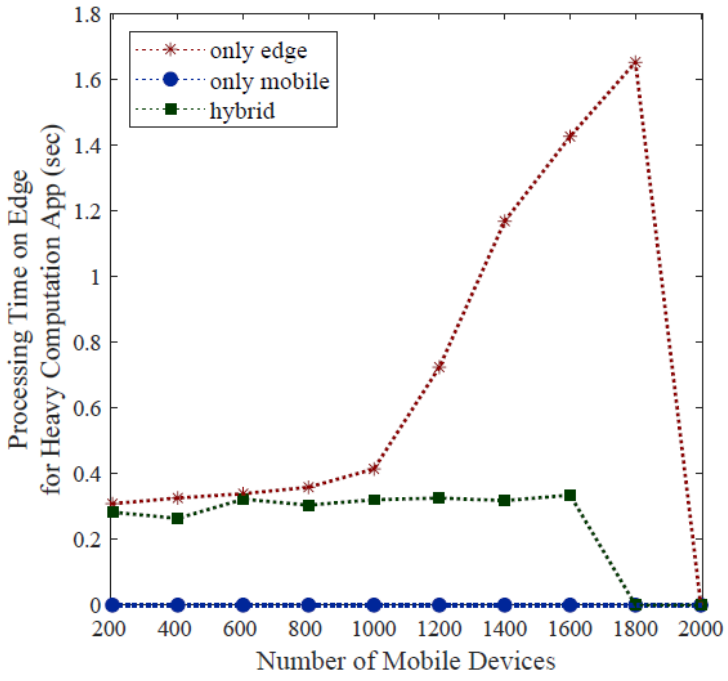


Figure 4.23: Processing time for Heavy Computation Application results

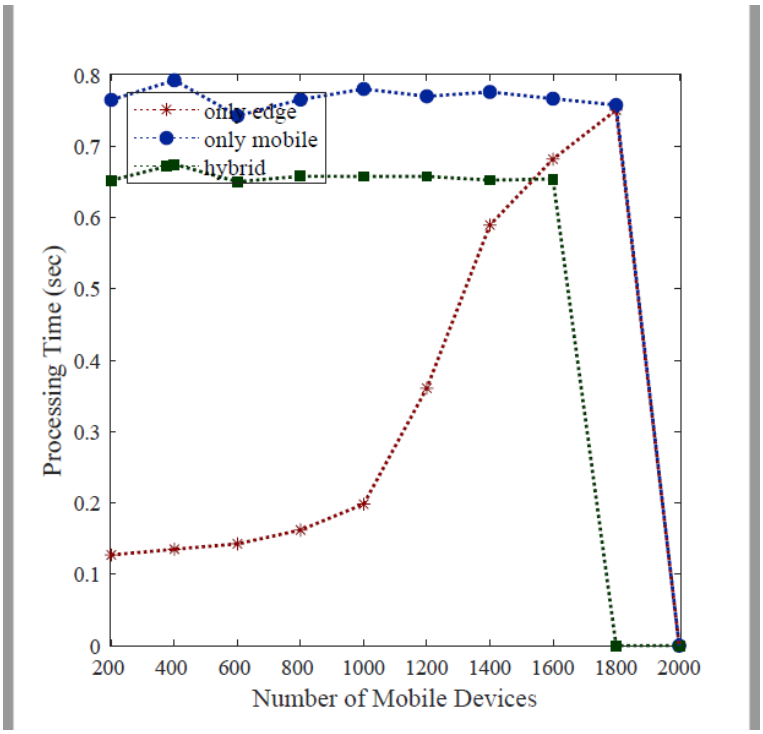


Figure 4.24: The average for Processing Time At The Edge Results

work since everything is processed locally. Most heavy and resource intensive applications are processed at the edge and sent later to the cloud. Hence the average network is high for edge only scenario as compared to the hybrid scenario.

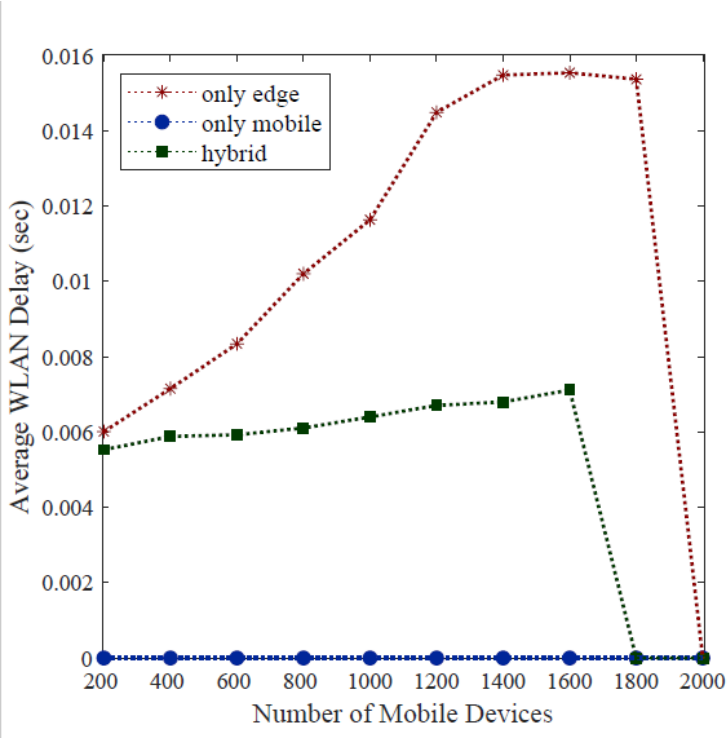


Figure 4.25: Average WLAN delay results.

Service time

Service time is time taken by the system to process the request. Figure 4.26 to figure 4.29 present results for service time for all the applications used in the simulated environment. This service time is evaluated with respect to the average size of a task. The researcher began by evaluating how each

applications performs in terms of service time when the task is offloaded to the Edge. It was observed that the service time is impacted by the average size of the task. Lightweight applications have lesser service time compared to resource intensive applications because of the complexity of task they offload. It is important also to note that the mobile device layer is zero because tasks are offloaded on Edge, as well as either edge or cloud (hybrid). The researcher evaluated the performance of the cloudlet network with our proposed Security model applied as shown in figure 4.30. The processing time at the Edge is lower compared to processing time of of mobile device and hybrid as expected, this is due to the fact that edge servers provide better performance for end users with decreased latency but the same cannot be said for cloud computing as the cloud server is far-end, providing good data storage with more latency [42]. These results also shows that the proposed security model does not negatively impact the network when it comes to service time.

Authors in[42] discuss the requirements that need to be met in order to successfully integrate blockchain with edge computing:

- Computational latency: This indicates the time spent on data processing and blockchain mining, which considers the computational power of system. In the study, block is mined by one node instead of entire nodes on the network resulting in a reduce computational power needed. As a result, computational power is reduced, hence service graph (Figure 4.30) shows that the blockchain has no negative impact on the performance of edge network.

- Authentication: In edge computing, with all data moving across the

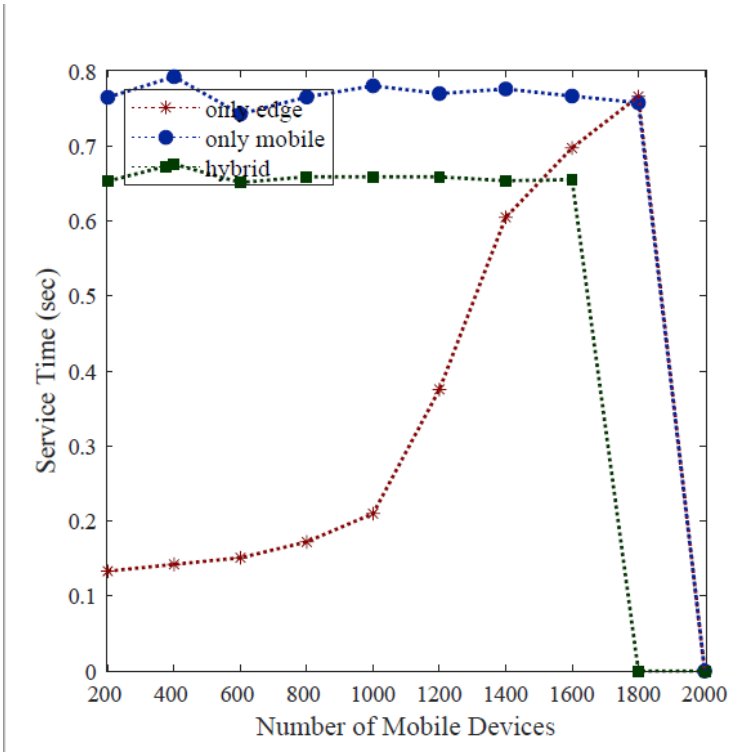


Figure 4.26: Service time on edge for Augmented Reality App results

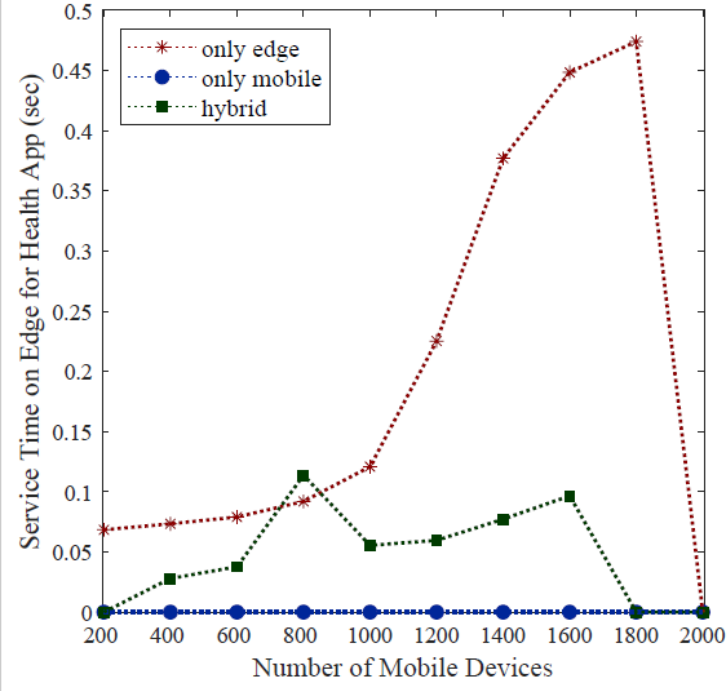


Figure 4.27: Service time on edge for Health App results

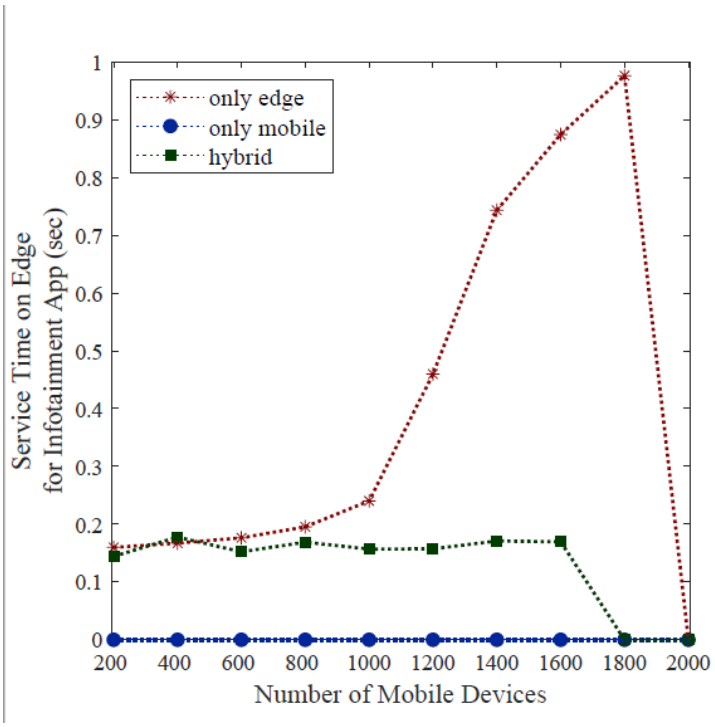


Figure 4.28: Service time on edge for Infoteinment App results

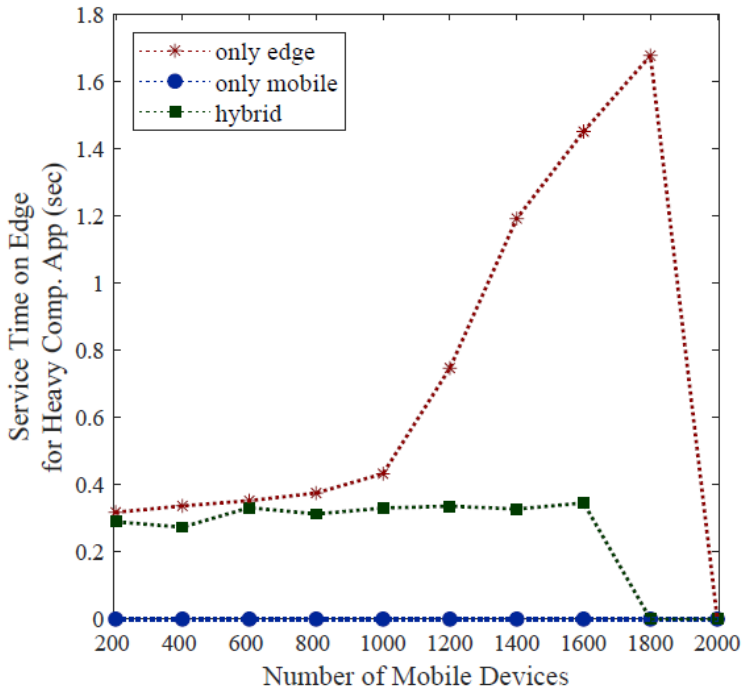


Figure 4.29: Service time on edge for Heavy Computation App

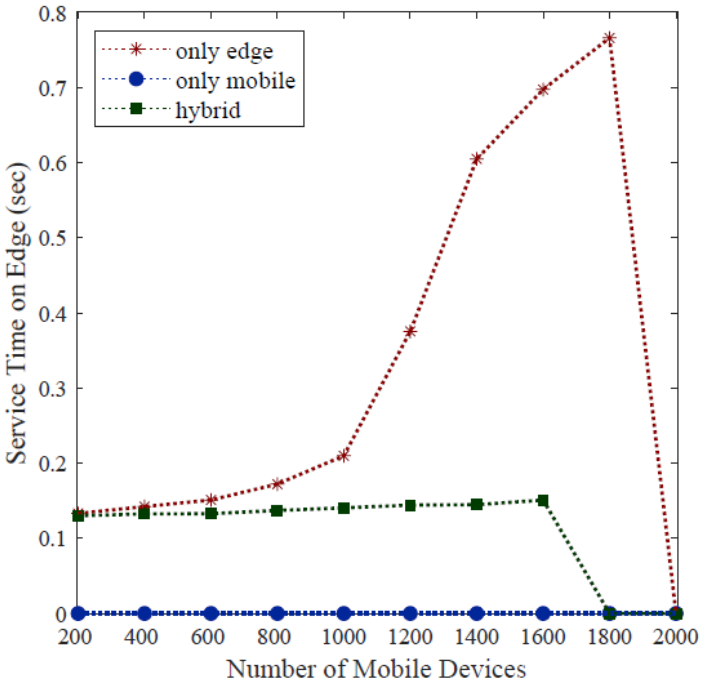


Figure 4.30: Average of Service time on edge results

layers and services, this raises the significance of authentication in this paradigm. In our study, we have implemented an Agent layer, between mobile device layer and Cloudlet layer that plays a paramount role of authenticating edge devices as well as controlling the stake of each node which is used in our Proof of Trust Consensus mechanism. This ensures authenticity of nodes partaking in mining and validating the chain. However, this did have a small effect on the Processing time shown in Figure 4.24 in cases where there is a disagreement between the nodes during validation as this process needs to be performed again.

4.3.3 Result Comparisons

In comparison of the proposed security model to other methods from literature which are used in securing data in the Edge computing environment, many dimension of comparison were identified, such as the Security model used, together with the Security scheme, the Consensus mechanism with interest if blockchain is used as a security mechanism or not, the number of nodes that participated in message verification and mining of the block for blockchain, and the security end goal- at what level is data being secure in terms of the Edge computing hierarchy.

Table 4.2 summaries the approaches used in literature. In [9], a blockchain-based cloudlet management method was proposed for multimedia applications. The aim was to ensure data integrity of data from the multimedia application's offloaded in cloudlet. They made use of objective function to measure QoS performance of multimedia applications which are modeled as multimedia workflows, and determine the scheduling strategy for these workflows. Each node in the network competes via Proof-of-Work to add the offloading information to a block and distribute the block to other nodes for validation. Once the block is validated, it is then added to the chain. Before the task is migrated by the cloudlet, transmitted data is compared to the scheduling information, if there is conflict then the migration is cancelled. However, in [27] and [9], the authors used proof of work consensus mechanism which requires intensive puzzle solving when mining a block, and also requires all the nodes(cloudlet) as shown in column named number of miners to participate in mining. The approach employed and implemented by the

authors has proven to be of high energy consuming mechanisms as the puzzles being solved are reported to be very resource intensive when it comes to resources utilization such as power.

In [3], the authors proposed a TrustChain which make use of blockchain as well as trust concept to overcome shortfalls such as privacy and energy consumption associated with traditional blockchain architectures in Edge Computing. Their proposed block chain demonstrates survivability as it makes use of low computing power and storage resource which is not the case for traditional blockchain technology. They designed a lightweight consensus management protocol by incorporating BFT protocol. This mechanism is only dependent on mutual agreements and trust amongst nodes rather than computing resources to mine a block. It also demonstrated an improved privacy due to intelligent encryption algorithm adopted inside the chain to eliminate data exposure.

In [28], the authors proposed an energy-efficient protocol called DEAN which ensures that data is not manipulated in edge nodes. The first step of execution of this protocol is selection of leaders amongst the nodes based on their trustworthiness. These leaders are responsible for authenticating the new node joining the network. The new node is given a block to validate and share the results with the leaders, based on these results, the leaders vote for the node trustworthiness. Another significant role of leaders is selecting one leader that will take part in validating the block before it is added on the chain. This study also take care of how the block is distributed in

the network, by proposing a smart sharing mechanism in which a block is distributed to adjacent nodes only with storage capacity such that when the node runs out of storage, an older block is moved to a closer node with enough disk capacity. However, a miner is selected based on their trustworthiness. They put trust to single node as only one node needs to be trusted to properly validate the message and mine the block. However, when this node is compromised, the entire network will be compromised as well. Therefore, authors implemented mechanism also failed to explore ways of verifying the trustworthiness of a node through out the mining process.

In this study, the researcher defined and used the implemented concept of proof of trust consensus mechanism, which uses elliptic curve method to verify a message from device to cloudlet. Moreover,the researcher avoided the possible security implications that comes with putting so much trust in a single node to remain trustworthy throughout the process. To achieve this concept in our proposal, We defined trust and experience based on the number of coins a node possesses, and two nodes are selected for verifying the incoming message, with the possibility of also addressing any future conflicts between the two nodes. This type of conflict is resolved by introducing a third node, performing the whole process of message verification again. Results were compared, the node with different results compared to the third node, loses all the coins and the others get awarded for their work. This ensures that our trust level is controlled and reliable. With the use of elliptic curve and no complex solution being performed, the model has minimal energy consumption, demonstrate capabilities of being secure with less power

consumption as previously proposed in Chapter 1, relative to the research aim and specific objectives.

Table 4.2: Comparison of results from securing data from edge computing.

Ref	**Security Model**	**Security Scheme**	**Consensus Mechanism**	**Number of Miners**	**Security Goal**
[3]	Blockchain	Digital Signature	Proof of Trust	Multiple nodes	Secure data at Edge layer
[6]	Blockchain	cryptography hash function	None	Single Node	Secure data at Cloud-Cloudlet level
[9]	Blockchain	None	Proof of Work	All nodes in the network	Protect Data stored in cloudlet
[13]	Data Security Protocol	Concept of Forward Secrecy , Session key	N/A	N/A	Protect access to data stored in the cloud from intruders and Cloud owner
[27]	Blockchain	Elliptic Curve Digital signature, DSSC, ISSC	Proof of Work	All nodes in the network	Protect Data stored in cloudlet
[28]	Blockchain	SHA-256 Encryption	Decentralized Edge Authorization Network	A single node the network	Protect Data stored in cloudlet, as well as reduced energy consumption blockchain security mechanism

4.4 Summary

In this chapter the researcher presented the implementation of the proposed security model in EdgeCloudSim which is the simulator developed for Edge Computing paradigms network simulation. The researcher further demonstrated how the implemented work can be tested by implementing a script for data modification. Furthermore, the study evaluated and recorded results for four applications(Health, Infotainment, Augmented Reality, and Heavy Computation). From the results, it is observed that the proposed security model has no negative impact on the network performance.

Chapter 5

5.1 Conclusion and future work

The advancement and forever changing technology has enable the development of sophisticated applications with remarkable possibilities of simplifying our complex daily activities. However, the limitations of the mobile devices such as limited battery lifespan makes it impossible to run and execute these applications. Cloud computing was then introduced, which allowed mobile devices to offload and run on the cloud. Due to the architecture of cloud computing network being centrally controlled, it suffers from latency when the number of mobile devices increases, and this negatively impact the performance of the network. The cloudlet computing platform was brought in to overcome cloud computing limitations by bringing cloud to the edge of the network. Its being closer to the user makes it susceptible to intruders and attacks. Hence, security model for data protection is needed.

The study findings demonstrate the design and development of a blockchain technology as security mechanism for cloudlet computing environment. The

study introduces a new consensus mechanism, proof of trust and an agent layer - between mobile device layer and cloudlet layer. In this introduced consensus mechanism, a miner is selected based on trust and experience which is determined by the number of coins each node has. This is because when a node makes any mistake in terms of data modification verification, it loses coins. Therefore, based on these concepts, the research can be able to determine how much trustworthy the node is as well as the experience of correct verification. Moreover, unlike the proof of stake consensus mechanism that uses all the miners participating in the mining tasks, which results in high energy consumption. The study's implemented proof of trust has two miners instead of one with the possibility of introducing additional one miner only if the two miners do not reach consensus. This approach has shown minimal energy consumption as compared to the previous method. Also, it does not put trust to a single node as this is too risky, and compared to the proof of stake mechanism discussed in section 2.2. All verified data is stored in blockchain in cloudlet by the miner and distributed to all other nodes.

The proposed security model can be improved by implementing Smart contract technique. Smart contract is a program that is stored in blockchain and runs when predefined conditions are met. This can be used to evaluate the validity of the chain without having to wait for new block to be added on the chain for validation to occur and isolate a compromised node. This will help in preventing attacks such as denial of service(Dos) at the application layer.

Bibliography

[1] Z. Pang, L. Sun, Z. Wang, E. Tian, and S. Yang, "A survey of cloudlet based mobile computing," in *2015 International Conference on Cloud Computing and Big Data (CCBD)*, pp. 268–275, IEEE, 2015.

[2] C. Sonmez, A. Ozgovde, and C. Ersoy, "Edgecloudsim: An environment for performance evaluation of edge computing systems," *Transactions on Emerging Telecommunications Technologies*, vol. 29, no. 11, p. e3493, 2018.

[3] U. Jayasinghe, G. M. Lee, MacDermott, and W. S. Rhee, "Trustchain: A privacy preserving blockchain with edge computing," *Wireless Communications and Mobile Computing*, vol. 2019, pp. 1–17, 07 2019.

[4] L. Khandare and D. S. Desai, "Analysis on privacy protection in cloudlet and edge technology," pp. 1–5, 09 2019.

[5] M. Babar, M. S. Khan, F. Ali, M. Imran, and M. Shoaib, "Cloudlet computing: Recent advances, taxonomy, and challenges," *IEEE Access*, vol. 9, pp. 29609–29622, 2021.

[6] A. B. Lazreg, A. B. Arbia, and H. Youssef, "Cloudlet-cloud network communication based on blockchain technology," in *2020 International Conference on Information Networking (ICOIN)*, pp. 164–169, IEEE, 2020.

[7] Z. Xiong, Y. Zhang, D. Niyato, P. Wang, and Z. Han, "When mobile blockchain meets edge computing," *IEEE Communications Magazine*, vol. 56, no. 8, pp. 33–39, 2018.

[8] H. Zeyu, X. Geming, W. Zhaohang, and Y. Sen, "Survey on edge computing security," in *2020 International Conference on Big Data, Artificial Intelligence and Internet of Things Engineering (ICBAIE)*, pp. 96–105, IEEE, 2020.

[9] X. Xu, Y. Chen, Y. Yuan, T. Huang, X. Zhang, and L. Qi, "Blockchain-based cloudlet management for multimedia workflow in mobile cloud computing," *Multimedia Tools and Applications*, vol. 79, no. 15, pp. 9819–9844, 2020.

[10] X. Xu, X. Liu, L. Qi, Y. Chen, Z. Ding, and J. Shi, "Energy-efficient virtual machine scheduling across cloudlets in wireless metropolitan area networks," *Mobile Networks and Applications*, vol. 25, no. 2, pp. 442–456, 2020.

[11] A. Tošić, J. Vičič, M. D. Burnard, and M. Mrissa, "A blockchain-based edge computing architecture for the internet of things," 2021.

[12] U. Jayasinghe, G. M. Lee, and A. MacDermott, "Trust-based data controller for personal information management," in *2018 International*

Conference on Innovations in Information Technology (IIT), pp. 123–128, IEEE, 2018.

[13] M. Jindal and M. Dave, "Data security protocol for cloudlet based architecture," in *International Conference on Recent Advances and Innovations in Engineering (ICRAIE-2014)*, pp. 1–5, IEEE, 2014.

[14] C. Quan and D. Qian-Ni, "Cloud computing and its key techniques," *Journal of Computer Applications*, vol. 29, no. 09, p. 2562, 2009.

[15] L. Wang, G. Von Laszewski, A. Younge, X. He, M. Kunze, J. Tao, and C. Fu, "Cloud computing: a perspective study," *New generation computing*, vol. 28, no. 2, pp. 137–146, 2010.

[16] A. Gajbhiye and K. M. P. Shrivastva, "Cloud computing: Need, enabling technology, architecture, advantages and challenges," in *2014 5th International Conference-Confluence The Next Generation Information Technology Summit (Confluence)*, pp. 1–7, IEEE, 2014.

[17] W. Shi, J. Cao, Q. Zhang, Y. Li, and L. Xu, "Edge computing: Vision and challenges," *IEEE internet of things journal*, vol. 3, no. 5, pp. 637–646, 2016.

[18] K. Al Nuaimi, N. Mohamed, M. Al Nuaimi, and J. Al-Jaroodi, "A survey of load balancing in cloud computing: Challenges and algorithms," in *2012 second symposium on network cloud computing and applications*, pp. 137–142, IEEE, 2012.

[19] K. Cao, Y. Liu, G. Meng, and Q. Sun, "An overview on edge computing research," *IEEE access*, vol. 8, pp. 85714–85728, 2020.

[20] M. Satyanarayanan, P. Bahl, R. Caceres, and N. Davies, "The case for vm-based cloudlets in mobile computing," *IEEE pervasive Computing*, vol. 8, no. 4, pp. 14–23, 2009.

[21] T. Verbelen, P. Simoens, F. De Turck, and B. Dhoedt, "Cloudlets: Bringing the cloud to the mobile user," in *Proceedings of the third ACM workshop on Mobile cloud computing and services*, pp. 29–36, 2012.

[22] U. Shaukat, E. Ahmed, Z. Anwar, and F. Xia, "Cloudlet deployment in local wireless networks: Motivation, architectures, applications, and open challenges," *Journal of Network and Computer Applications*, vol. 62, pp. 18–40, 2016.

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

[24] S. M. S. Saad and R. Z. R. M. Radzi, "Comparative review of the blockchain consensus algorithm between proof of stake (pos) and delegated proof of stake (dpos)," *International Journal of Innovative Computing*, vol. 10, no. 2, 2020.

[25] R. Chatterjee and R. Chatterjee, "An overview of the emerging technology: Blockchain," in *2017 3rd International Conference on Computational Intelligence and Networks (CINE)*, pp. 126–127, 2017.

[26] S. Bouzefrane, A. F. B. Mostefa, F. Houacine, and H. Cagnon, "Cloudlets authentication in nfc-based mobile computing," in *2014 2nd*

IEEE International Conference on Mobile Cloud Computing, Services, and Engineering, pp. 267–272, IEEE, 2014.

[27] J. Kang, R. Yu, X. Huang, M. Wu, S. Maharjan, S. Xie, and Y. Zhang, "Blockchain for secure and efficient data sharing in vehicular edge computing and networks," *IEEE Internet of Things Journal*, vol. 6, no. 3, pp. 4660–4670, 2018.

[28] A. Al-Mamun and D. Zhao, "Trustworthy edge computing through blockchains," *arXiv preprint arXiv:2005.07741*, 2020.

[29] G. Qiao, S. Leng, H. Chai, A. Asadi, and Y. Zhang, "Blockchain empowered resource trading in mobile edge computing and networks," in *ICC 2019-2019 IEEE International Conference on Communications (ICC)*, pp. 1–6, IEEE, 2019.

[30] S. Guo, X. Hu, S. Guo, X. Qiu, and F. Qi, "Blockchain meets edge computing: A distributed and trusted authentication system," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 3, pp. 1972–1983, 2019.

[31] E. Bonnah and J. Shiguang, "Decchain: A decentralized security approach in edge computing based on blockchain," *Future Generation Computer Systems*, vol. 113, pp. 363–379, 2020.

[32] L. Yuan, Q. He, S. Tan, B. Li, J. Yu, F. Chen, H. Jin, and Y. Yang, "Coopedge: A decentralized blockchain-based platform for cooperative edge computing," in *Proceedings of the Web Conference 2021*, pp. 2245–2257, 2021.

[33] X. Xu, X. Zhang, H. Gao, Y. Xue, L. Qi, and W. Dou, "Become: Blockchain-enabled computation offloading for iot in mobile edge computing," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 6, pp. 4187–4195, 2019.

[34] Y. Ren, Y. Leng, Y. Cheng, and J. Wang, "Secure data storage based on blockchain and coding in edge computing," *Math. Biosci. Eng*, vol. 16, no. 4, pp. 1874–1892, 2019.

[35] M. Zhaofeng, W. Xiaochang, D. K. Jain, H. Khan, G. Hongmin, and W. Zhen, "A blockchain-based trusted data management scheme in edge computing," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 3, pp. 2013–2021, 2019.

[36] S. Bouzefrane, A. F. B. Mostefa, F. Houacine, and H. Cagnon, "Cloudlets authentication in nfc-based mobile computing," in *2014 2nd IEEE International Conference on Mobile Cloud Computing, Services, and Engineering*, pp. 267–272, 2014.

[37] M. Arif, F. Ajesh, S. Shamsudheen, and M. Shahzad, "Secure and energy-efficient computational offloading using lstm in mobile edge computing," *Security and Communication Networks*, vol. 2022, 2022.

[38] K. Bhardwaj, J. C. Miranda, and A. Gavrilovska, "Towards {IoT-DDoS} prevention using edge computing," in *USENIX Workshop on Hot Topics in Edge Computing (HotEdge 18)*, 2018.

[39] U. Jayasinghe, G. M. Lee, T.-W. Um, and Q. Shi, "Machine learning based trust computational model for iot services," *IEEE Transactions on Sustainable Computing*, vol. 4, no. 1, pp. 39–52, 2018.

[40] Y. Huang, J. Zhang, J. Duan, B. Xiao, F. Ye, and Y. Yang, "Resource allocation and consensus on edge blockchain in pervasive edge computing environments," in *2019 IEEE 39th international conference on distributed computing systems (ICDCS)*, pp. 1476–1486, IEEE, 2019.

[41] C. Sonmez, A. Ozgovde, and C. Ersoy, "Performance evaluation of single-tier and two-tier cloudlet assisted applications," in *2017 IEEE International Conference on Communications Workshops (ICC Workshops)*, pp. 302–307, 2017.

[42] R. Yang, F. R. Yu, P. Si, Z. Yang, and Y. Zhang, "Integrated blockchain and edge computing systems: A survey, some research issues and challenges," *IEEE Communications Surveys & Tutorials*, vol. 21, no. 2, pp. 1508–1532, 2019.

[43] Z. Du, C. Wu, T. Yoshinaga, K.-L. A. Yau, Y. Ji, and J. Li, "Federated learning for vehicular internet of things: Recent advances and open issues," *IEEE Open Journal of the Computer Society*, vol. 1, pp. 45–61, 2020.