Debre Birhan University College of Computing
Department of Information Technology
MSC in Computer Networks and Security

Blockchain Cryptographic -based Authentication and Verification for Secure Communication in Mobile Ad-hoc Network (MANET)

A thesis submitted to the Department of Information Technology in partial fulfillment of the requirements for the degree of Master of Science in Computer Networks and Security

BY: Zenebech Belete

Advisor: Samuel Asferaw (Ph.D.)

Debre Birhan Ethiopia

July 2021

Debre Birhan University College of Computing
Department of Information Technology
MSC in Computer Network and Security

Blockchain Cryptographic -based Authentication and Verification for Secure Communication in Mobile Ad-hoc Network (MANET)

A Thesis Submitted in Partial Fulfilment of the Requirements for The Degree of Master of Science in Computer Networks and Security

Zenebech Belete Endashaw

Advisor: Samuel Asferaw (PhD)

Debre Birhan Ethiopia,

July 2021

Debre Birhan University
College of Computing Sciences
Department of Information Technology

Approval page

Zenebech Belete Endashaw

Advisor: Samuel Asferaw (Ph.D.)

This is to certify that the thesis prepared by Zenebech Belete Endashaw, titled: Blockchain Cryptographic -based Authentication and Verification for Secure Communication in Mobile Ad-hoc Network (MANET) and submitted in partial fulfillment of the requirements for the Degree of Master of Science in Computer Networks and Security.

Approved by:

	Name	**Signature**	**Date**
Advisor:	Samuel Asferaw (Ph.D.)	_____________	_________
Internal Examiner:	________________	________________	___________
External Examiner:	________________	_______________	___________
Chai Person	________________	_______________	___________

Zenebech Belete Endashaw

Dedication

This work is Dedicated to:

My MOM (Welansa Negesse) and

My Children (Tinsae, Amen, and Eyoase)

Declaration I, the undersigned, hereby declare that this thesis is my original work performed under the supervision of **Samuel Asferaw (Ph.D.),** has not been presented as a thesis for a degree program in any other university, and all sources of materials used for the thesis are duly acknowledged.

Name: Zenebech Belete

Signature: ____________

Place: Debre Birhan

This thesis has been submitted for examination with my approval as a university advisor.
Samuel Asferaw (Ph.D.)

Advisor Signature _________________

ABSTRACT

One of the most popular wireless network technologies is mobile ad-hoc networks (MANET). A MANET network is a decentralized, self-organizing, and infrastructure-less network. In MANET network there is no any administrative node that controls the entire network, every node participating in the network is responsible for the reliable operation of the network and the activities of the network. Due to the nature of MANET's network characteristics, the key issues to design the routing protocol are security and network performance. In terms of network performance, AODV has better performance than other MANET routing protocols. However secure routing is an essential part of MANET for protecting the network operations from malicious nodes.

To address this secure routing problem in MANET, we proposed a new secure AODV protocol called blockchain-based authentication and verification security mechanism for securing MANET communication. A blockchain security mechanism is used mainly in the financial institution to secure financial transactions. But now a day's, many business organizations use a Blockchain system for securing their asset and communication. Communication participants in blockchain-based authentication and verification use SHA1 and SHA5 cryptographic techniques used to make a block of the messages. Every node in the network must have the key table to authenticate and verify the neighbor node incoming data. If a node is suspected as an attacker, the security mechanism will isolate it from the network before communication is established and put the node in the rejected list.

The performance of the proposed method is evaluated under malicious nodes by using network simulator 2 (NS-2). It is compared with a symmetric key authentication routing protocol. Security performance is evaluated in terms of Detection Rate (DR), False Positive Rate (FPR), and False Negative Rate (FNR), and also we evaluated network performance in term of end to End Delay (E2ED), Packet Delivery Ratio (PDR), and throughput (TH). The simulation result of our proposed algorithm shows that the detection rate of the proposed system is 87.76% and the proposed method has better network performance than the symmetric key authentication method in terms of packet delivery ratio and throughput, but in terms of end-to-end delay, there is no improvement. The average packet delivery ratio increases by 7.98%, and the average throughput increases by 4.61%. However, the average end-to-end delay value decreases by 0.031%.

Keyword: Blockchain, MANET, security, AODV, Performance,

Acknowledgments

First, I would like to thank almighty GOD who gave me the power, patience, health, environment, and people to support me to complete my study and my thesis work.

In addition, I would like to express my deepest feeling towards my Adviser, Samuel Asferaw (Ph.D.), for all his advice, valuable support, guidance, constructive comments, patience, time, and knowledge, which he provided during my study and my thesis work.

Finally, I thank all the family especially my MOM. Since it is difficult to reference my friend’s and classmate’s role to my achievements in words, it is well to say I recorded it throughout my life in my heart.

Table of Contents

List of Figures

List of Tables

List of Abbreviations

AODV	Ad-hoc on-demand Distance Vector
AWK	Aho, Weinberger, and Kernighan Scripts
BAN	Body Area Network
CBR	Constant Bit Rate
DoS	Denial of Service
DDOS	Distributed Denial of Service Attack
DSA	Digital Signature Algorithm
DSDV	Destination-Sequenced Distance Vector
DSR	Dynamic Source Routing
DSS	Digital Signature Standard
E2E	End-to-end delay
HC	Hope Count
IEEE	Institute of Electrical and Electronics Engineers
MAC	Medium Access Control
MANET	Mobile Ad-hoc Network
NAM	Network Animator
NIST	National Institute for Standards and Technology
MD	Message Digest
NS-2	Network Simulator-2
OTCL	Object-oriented Tool Command Language
OLSR	Optimized Link-State Routing

PDR	Packet Delivery Ratio
PO	Post Office
RERR	Route Error
RREP	Route Reply
RREQ	Route Request
RSU	Road Side Units
SHA	Secure Hash Algorithm
SAR	Security-Aware Ad-hoc Routing
SHS	Secure Hash Standard
TCL	Tool Command Language
TCP	Transmission Control Protocol
TBRPF	Topology-Based Reverse Path Forwarding
UDP	User Data Gram Protocol
UAV	Unmanned Airborne Vehicles
UWB	Ultra-Wide Band
VANET	Vehicular Ad-hoc Network
WSN	Wireless Sensor Network
ZRP	Zone Routing Protocol

Chapter One

Introduction

1.1 Research Background

In the wireless environment, a connection between devices should be established instantly and should self-configure all the time as the devices may move in random directions all the time since the wireless range is limited. These properties are present in ad-hoc wireless networks. From wireless ad-hock network classification, a mobile ad-hoc network (MANET) is one of the most usable wireless connection features. Mobile ad-hoc network (MANET) is a peer to peer ad-hoc network which is characterized by having dynamic and random topology set up, multi hope routing, wireless connectivity, no central administration, rapid mobility of node, shared medium, and self-possessed of some constraints like low bandwidth, high power consumption, and limited storage capacity of the devices used in the network. The main purpose of the mobile ad-hoc network is to support strong and efficient operation in a mobile wireless network by incorporating routing functionality and securing data sharing into mobile nodes. Many scholars are working on the above MANET issues. MANET can be either connected to large internet or operate in a stand-alone mode which is formed locally between mobile nodes [1].

MANET is considered a robust and scale-able network infrastructure, and it concerns several areas such as security, availability, reliability, and random network formation. MANET is a crucial research topic and requires a completely different approach of analysis than the wired networks. Mobile ad-hoc networks are susceptible to the malicious behavior of nodes. The absence of certification authority and centralized infrastructure administration are the main cause of MANET vulnerability to a security problem. Nodes in MANETs can enter or leave anytime the network and there are chances that channels transfer data through nodes that are not genuine. There should be some criteria to define the membership of nodes in MANETs and the basis of classifying trusted and non-trusted nodes. The methods presently in use for securing data sharing by cryptographic techniques are described reference [2].

The blockchain concept originated as a solution to electronic currency transfer without necessitating a central authority of the bank system. Blockchain use peer to peer network which is responsible to ensure free communication among the blockchain nodes, where the nodes are geographically distributed in a different location but being equal participants in the application of the network. In a blockchain, there is no centralized server and each node has information of other users, but also an information provider to other nodes who join the network lately. Each node in the network involves in the routing process of the entire network, the discovering and maintaining of connections to neighboring peers, the propagation and verification of transactions, as well as the synchronization of data blocks [2].

To days' Mobile ad-hoc network (MANET) security solutions are being adopted rapidly but result in vulnerabilities and exposure to different types of attack which is investigated by many scholars. Blockchain is the fundamental technology of Bitcoin which is the first cryptocurrency system and addressed security issues in the network system [3]. MANET is much more vulnerable to different types of attack compared to infrastructure-based networks his study mainly focuses on the security of MANET. Blockchain is suitable for a decentralizing system without central administration. We have studied the security which is improved by the principles of blockchain to find a relevant solution increasing the security level of mobile ad-hoc networks (MANET).

1.2 Statement of Problem

There are many critical issues and challenges in MANET i.e. network security and network performance [4]. The unique characteristics of MANET present a new set of serious and essential challenges to security design and the performance of the network these include open peer-to-peer network architecture, shared wireless medium, resource constraints, and extremely dynamic network topology. These types of challenges make a case for creating security and performance solutions that achieve both deep protection and desirable network security and performance. MANET has different personalities and characteristics that surely causing their specific security and network performance concerns. Since MANET has high mobility of nodes, no administrative node to control the network and it uses an open network which is nodes can join the network at any time and leave the network at any time. Every node can participate in the network operation like route formation and data sharing so this makes MANET more vulnerable to malicious attacks.

MANET is vulnerable for different types of attacks which listen, read, modify and drop the data transferred between nodes. All types of attacks are happening in MANET is because of the formation and nature of the network. In MANET the network is formed if nodes are willing to share data. So MANET has no infrastructure, no central administration, and has the dynamic configuration of the network. It is difficult to control who is in the network and protect the data sharing between the communicating nodes of MANET. Many potential attacks can be performed in each communication layer of the network. MANET is more vulnerable to several attacks such as spoofing, eavesdropping and Denial of Service (DoS), and any other types of attacks.

The security and network performance challenges in the MANET arise due to its dynamic topology, open wireless link, and mobility of nodes. An identification mechanism is needed between the nodes to accept the node and reject the node. Current routing protocols do not focus much on security and confidentiality issues. These aspects are suspended and need further development. An authentication and verification protocol is needed between nodes using some cryptographic method. Much MANET security-related work is still pending and will add to the standards as the physical deployment of the MANET network. Under these constraints, the routing protocol challenge in MANET is how to develop a strong security-aware routing protocol that will reduce the attacks in MANET without consuming the overall network performance. In this work, we develop a solution for routing protocol to cover the security performance and network performance problem in MANET. Our proposed security mechanism is Blockchain-based authentication and verification which identifying who joins the network by authentication of nodes and verify the message from where it comes from.

Hence, this research work tries to answer the following research questions:

1. How can we use blockchain cryptography method for securing MANET?
2. How can we develop a blockchain-based cryptographic algorithm for advancing security in MANET?
3. What are the network security and network performance metrics used for evaluation of the proposed work?

1.3 Research Objective

1.3.1 General Objective

The general objective of this study is designing an algorithm which secure the communication between nodes in Mobile ad-hoc network using blockchain cryptographic -based authentication and verification method.

1.3.2 Specific Objectives

The followings are some lists of specific objectives:

- ✓ Develop Blockchain cryptographic-based authentication and verification algorithm for secure the communication in MANET
- ✓ Implement the developed algorithm by using NS-2 simulator
- ✓ To select appropriate simulation tool and performance metrics
- ✓ Evaluate the proposed algorithm with the existing system by selected evaluation metrics

1.4 Scope and Limitation

The scope of this study is mainly focused on securing mobile ad-hoc networks (MANET) by blockchain cryptographic-based authentication and verification methods. We evaluate the proposed solution using NS-2 simulation in the ad-hoc network topology. And security performance is evaluated in terms of Detection Rate(DR), False Positive Rate (FPR), and False Negative Rate (FNR), and also we evaluate network performance in terms of end to end delay (E2ED), packet delivery ratio (PDR), and Throughput (TH).

Because of time and other resource constraints this study faces limitation on implementation of different types of attack on the proposed algorithm. It only compares with the existing work which Symmetric Key Based Authentication Mechanism which is base paper of this study.

1.5 Significance of the Study

The main significance of this study are: -

- Improving security performance and network performance of Mobile ad-hoc network (MANET)
- We propose a new security mechanism to secure the communication of nodes in a mobile ad-hoc network (MANET)
- This research work would be used as a reference for further study

1.6 Contributions of the Study

Contributions of this thesis work are:

- ✓ Develop a new security mechanism to securing the MANET by modifying the AODV routing protocol.
- ✓ Implemented Blockchain-based authentication and verification algorithm to improve the AODV routing protocol.
- ✓ Improved the security and network performance of MANET

1.7 Thesis Organization

The remaining parts of this paper are organized as follows chapter two presents literature review and related works. Chapter three describes the developed algorithm which is blockchain cryptographic-based authentication and verification. Chapter four deals with simulation design and performance metrics of Block-Chain Cryptographic -based Authentication and Verification. Chapter five describes Simulation Result Performance Evaluation and Discussion of the developeded algorithm and finally Chapter six deals with the conclusion of the work and future work of the paper.

Chapter Two

Literature Review

2.1 Survey of Related Literature

2.1.1 Mobile ad-hoc Network (MANET)

A mobile ad-hoc network (MANET) is a collection of mobile nodes that can communicate with each other using wireless links. It is also possible to have access to some nodes in fixed infrastructure, depending on the kind of mobile ad-hoc network available. Some scenarios where an ad-hoc network can be used in business associates for sharing information during a meeting, an emergency disaster like storm, earthquake, or flooding. In this environment, a route between two nodes or hosts may consist of hops through one or more nodes in the MANET [5]. An essential problem in a mobile ad-hoc network is finding and maintaining routes since node mobility can cause topology changes and securing the data sharing between nodes. Several routing algorithms for MANETs have been proposed in the literature.

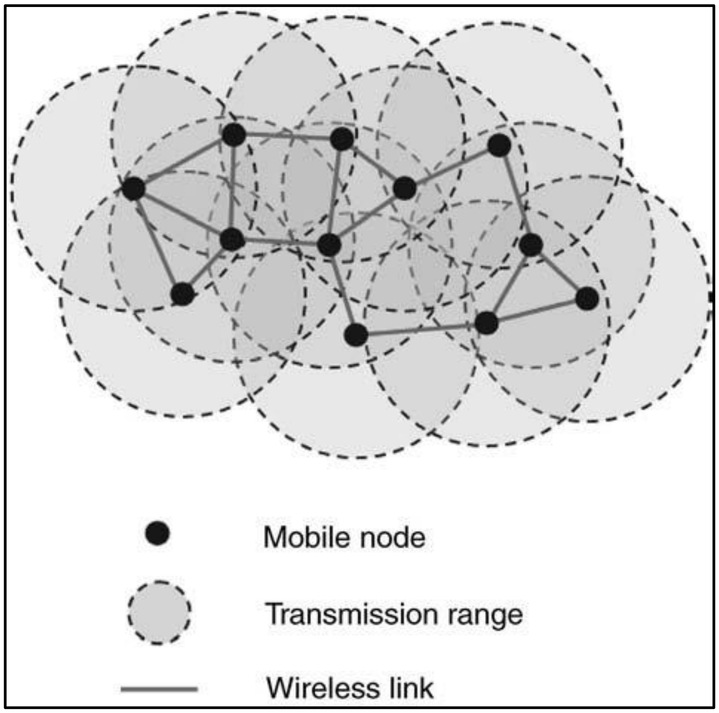


Figure 2-1: MANET Topology [5]

Mobile ad-hoc networks can be classified by different networks such as body area networks (BAN), vehicular ad-hoc networks (VANET), wireless networks, and wireless sensor networks (WSN). Moreover, MANETs can be realized by different wireless communication technologies such as Bluetooth, IEEE 802.11, and Ultra-Wide Band (UWB) [6].

A mobile ad-hoc network (MANET) is a self-governing ad-hoc wireless networking system consisting of independent nodes that the movement of a node is dynamically changing network connectivity. There is no fixed infrastructure exists in the MANET network, and no centralized administration can be available. The network can be formed anywhere, at any time, and formed if two or more nodes are connected and communicate with one another either directly when they are in the radio range of each other or via intermediate mobile nodes. MANET networks are planned to have dynamic, sometimes rapidly-changing topology, multi-hop routing which is likely composed of relatively bandwidth-constrained wireless links. The mobile nodes can perform the roles of both hosts and routers. The presence of mobility makes a MANET challenging for designing and implementation of security mechanisms in real life [7].

In real-world applications of MANET network are in the military, vehicular communications, disaster relief, emergency operations for free intercommunication between laptops and PC in a local area or small businesses, and delay-tolerant networking. With the arrival of newer technologies, mobile ad-hoc networks are becoming an integral part of next-generation networks because of their flexibility, auto-configuration capability and lack of infrastructure, ease of maintenance, self-administration capabilities, and cost-effectiveness.

MANET nodes within one another's wireless transmission range can communicate directly but, nodes outside one another's transmission range have to rely on some other nodes to transmit messages. Therefore, most of the time MANET uses a multi-hop scenario, and several intermediate nodes transmit the packets sent by the source node to make them reach the destination node.

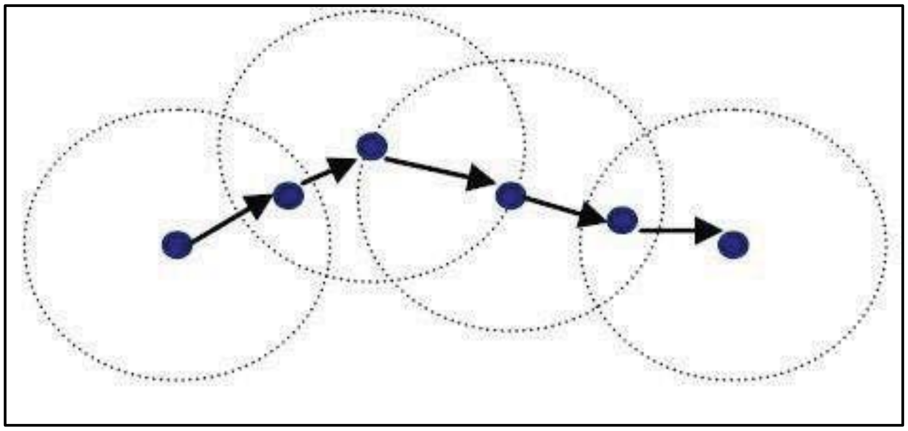


Figure 2-2: Communication in ad hoc network [7]

The dynamic nature of MANETs makes the network open to attacks and unreliability. Routing and secure routing are always the most significant part of any network. Each node should not only work for itself but should also be cooperative with other nodes as a route between the source and the destination. MANETs are vulnerable to various security attacks. Hence, finding a secure and trustworthy end-to-end path in MANETs is the challenging part [7].

In general, routing algorithms for ad-hoc networks may be divided into two broad classes: proactive protocols and reactive protocols, as discussed as follows.

Proactive Routing Protocols

Proactive routing algorithms aim to keep consistent and up-to-date routing information between every pair of nodes in the network by proactively propagating route updates at fixed time intervals. In proactive routing, each node maintains this information in tables and this protocol is called table-driven algorithms. Examples of proactive protocols are Destination-Sequenced Distance Vector (DSDV) [19], Optimized Link-State Routing (OLSR) [20], and Topology-Based Reverse Path Forwarding (TBRPF) Protocols [12].

Reactive Routing Protocols

Reactive on-demand routing algorithms establish a route to a given destination only when a node requests a route discovery process. Once a route has been established, the node keeps it until the destination is no longer accessible, or the route expires. Examples of reactive protocols are Dynamic Source Routing (DSR) and Ad-hoc On-Demand Distance Vector (AODV) [12]. The DSR protocol determines the complete route to the destination node, expressed as a list of nodes of the routing path, and embeds it in the data packet. Once a node receives a packet it simply forwards it to the next node in the path. DSR keeps a cache structure (table) to store the source routes learned by the node.

The AODV protocol keeps a routing table to store the next hop routing information for destination nodes. Each routing table can be used for a given time. If a route is not requested within that period, it expires and a new route formed when needed. Each time a route is used, its lifetime is updated. When a source node has a packet to send for a given destination, it looks for a route in its route table. In case there is one, it uses it to transmit the packet. Otherwise, it initiates a route discovery procedure to find a route by broadcasting a route request (RREQ) message to its neighbors. Upon receiving an RREQ message, a node performs the following actions: checks for duplicate messages and discards the duplicate ones creates a reverse route to the source node (the node from which it received the RREQ is the next hop to the source node) and checks whether it has an unexpired and more recent route to the destination by comparing to the one at the source node. In case those two conditions hold, the node replies to the source node with an RREP message containing the last known route to the destination. Otherwise, it retransmits the RREQ message [4]. The AODV protocol is briefly discussed in chapter four.

2.1.1.1 Characteristics of MANETs

- **No Infrastructure:** Mobile Ad-hoc networks have no pre-established fixed infrastructure. There is no predefined infrastructure for MANET.
- **Dynamic typologies**: nodes have the freedom to move randomly. The network topology typically changes randomly and rapidly at unpredictable times and may consist of both bidirectional and unidirectional links.
- **Decentralized Control**: Due to dynamic topology, the operation of MANET relies on the collaboration of members of the node.
- **Bandwidth-constrained**: MANETs have brought down limits and shorter transmission ranges. Wireless communication will continue to have significantly lower capacity than its hardwired counterparts.
- **Energy-constrained operation**: all of the nodes in a MANET may depend on batteries or use other means for energy. For these the most important system design criteria for MANET is optimization in energy saving.
- **Limited physical security**: Mobile wireless networks are susceptible to physical security threats. The increased possibility of eavesdropping, spoofing, and denial-of-service attacks should be carefully considered.

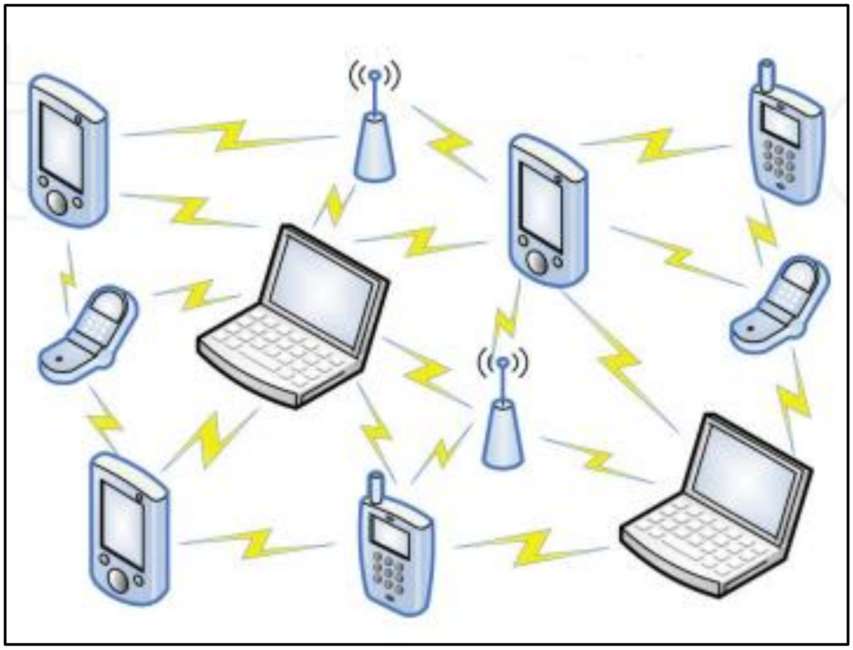


Figure 2-3: MANET Network [13]

2.1.1.2 Application Areas of MANET

Mobile ad-hoc networks have been working in situations where infrastructure is physically formed. In all these cases, there is a need for common computing and communication among the mobile users who typically work as teams like medical personnel in a search and rescue mission, firefighters facing a hazardous emergency, policemen conducting surveillance of suspects, and soldiers engaging in a fight. When we consider all these usual driving applications managed by specialized people, we understand why there is slow progress in deploying commercial ad-hoc applications to ordinary people [20].

A MANET can be used to provide access to crisis management applications, for example in a disaster recovery, where the entire communication infrastructure is destroyed and establishing communication quickly is crucial [12]. so we can simply form an ad-hoc network in that area using hours rather than days or weeks like the case of wired networking. When a user wants to use an existing application on the Internet in a mobile ad-hoc network, it is important to consider its performance.

Another application area is communication and coordination in a battlefield using self-governing networking [13]. Some military ad-hoc network applications require unmanned and robotic components. Unmanned Airborne Vehicles (UAVs) can cooperate in maintaining a large ground mobile ad-hoc network interconnected despite physical obstacles, propagation channel irregularities, and enemy jamming. The UAVs can help meet tight performance constraints on demand by proper positioning and antenna beaming.

Another application of the mobile ad-hoc network is a vehicular ad-hoc network (VANET) which is designed to provide communications among nearby vehicles and between vehicles and nearby fixed equipment. The main goal of a VANET is to provide safety and comfort for passengers [30, 31].

Generally, with many emerging applications, adaptable ad-hoc networks have the potential to allow a large number of devices to communicate end-to-end without requiring any pre-existing infrastructure and are very suitable to support general networking scenarios.

2.1.1.3 MANET Security Attacks

Mobile ad-hoc networks are generally more lying to physical security threats than fixed-wired networks [15, 16]. The transmission nature of the wireless channels, the absence of a fixed infrastructure, the dynamic network topology, the collaborative multi-hop communication among nodes, and the self-organizing characteristic of ad-hoc networks increase the vulnerabilities to attackers.

The starting point to provide a proper security solution for a mobile ad-hoc network is to understand the possible forms an attack can happen. In a MANET, a security problem may happen at any network layer and the attacks may include:

- ✓ Data integrity attacks
- ✓ Accessing the route information attacks
- ✓ Modification of the message attacks
- ✓ Injecting traffic attacks
- ✓ Denial-of-service attacks
- ✓ Flow-disruption attacks
- ✓ Delaying the network attacks
- ✓ Dropping, or corrupting data packets attacks
- ✓ Eavesdropping attack
- ✓ Signaling attacks or signal jamming attack
- ✓ Divert network traffic, or making routing inefficient attacks are the main types of attacks which are happened in mobile ad-hoc networks.

From the above-listed variety of possible attacks to a mobile ad-hoc network, different solutions have been proposed to address them [16]. The first step is to protect the wireless network infrastructure against malicious attacks. Digital signatures can be used to authenticate a message and prevent attackers from injecting erroneous routing information and data traffic inside the network [17]. This scheme requires a certification authority function to manage the private and public keys to distribute keys by certificates, which needs to be distributed over multiple nodes in the MANET [18]. In many mobile ad-hoc network applications, such as emergency disaster relief and information sharing in a meeting, it is important to guard against attacks such as malicious routing misdirection [19]. The problem is that ad-hoc routing protocols were designed to trust all participants, are cooperative by nature, and depend on adjacent nodes to route packets to the destination node.

Some of the proposed solutions to the problem of secure routing in a MANET involve the use of a pre-deployed security infrastructure [10], making the network topology or structure as in the Zone Routing Protocol (ZRP) [9], and introducing mechanisms in the network to mitigate routing misbehavior such as the Security-Aware Ad-hoc Routing (SAR) technique to add security attributes to the route discovery path of the network [11].

When communication establishes between two mobile devices, the networks must provide security and privacy. The security in MANETs is a major concern therefore communication between the nodes in MANET should be carried out insecure manner. The communication among the mobile users in MANETs needs to be more secure. Due to the lack of any stringent security policy and centralized management, it opens doors for the attackers and they can simply exploit attacks on nodes and the services provided by them. Because of vulnerabilities for different types of attacks, security in MANET is a challenging one.

All the above attacks occur due to the following reasons:

- Absence of installed infrastructure: there is no central administration of mobile ad-hoc network which administers certification and authentication process.
- Dynamically change of network topology: the dynamic changing of the mobile node or any time mobility of node make the security of routing protocols at risk.
- Power and computational limitations: because of battery and computational limitations we cannot use complex encryption algorithms for securing the mobile nodes.
- Multi-hop communication: it leads to create multiple communication among nodes over the network and nodes cannot know who is a genuine node and malicious node

2.1.1.4 Security Requirement for MANETs

Security requirements for the mobile ad-hoc networks are Authentication, confidentiality, non-repudiation, access control, and availability. These are the main criteria to define the capability of any network security. To make full the security requirements of MANET we develop the blockchain cryptographic authentication and verification method.

Authentication

Using authentication is possible to guarantee communicating parties' reliability and identify a sender rightly. In node communication, we can use authentication and security layer standards.

Integrity

To make sure the accuracy of transmitted information without modification, it is essential to limit data modification and prevent the problem to stability so integrity is the protection of data states correct from sender to the receiver node.

Confidentiality

Confidentiality is information protection from unauthorized read and access. Information should only be seen and accessed by those persons authorized to see and access it. In other words, it is avoiding reading and accessing the information by an illegitimate node.

Non-repudiation

Non-repudiation is a legal concept that is widely used in information security and refers to a service that provides proof of the origin of data and the integrity of the data. In other words, non-repudiation makes it very difficult to successfully deny who and where a message came from as well as the authenticity and integrity of that message.

Access control

Network access control allows you to identify who, what, where, when, and how an end-user or device is accessing the network and network resources.

2.1.2 Blockchain

The history of digital currency started in the 1990s by David Chaum an American cryptographer. He created the first online money transaction which is called DigiCash in the Netherlands. DigiCash use as an extension of the RSA (Rivest, Shamir, and Adleman) encryption algorithm. It became popular and Microsoft Corporation tried to buy DigiCash for $180 million. One of the crucial mistakes Chaum and his company made was to reject Microsoft's $180 million offer and earn the annoyance of the Netherland's primary monetary authority. All of those crucial mistakes eventually led to the expiration of DigiCash in 1998. The second internet-based money transaction is PayPal and its achievement is using credit cards and can transfer money to and from merchants and buyers, it's the most popular means by which to make transactions online. The next significant event in the history of cryptocurrencies is 2008 and the way to the rise of the blockchain, which is the foundation of cryptocurrencies [8].

In 2009, Satoshi Nakamoto published a white paper that illustrated the source code, technology, and concept of the blockchain and Bitcoin which is the first cryptocurrency. Today, there are more than 70 million units of Bitcoin that are circulating in the digital financial system and these have a total market capitalization of around $50 billion. These days there are more than 850 cryptocurrencies in the digital financial system being transacted internationally, which include Ethereum (Ether), Ripple, Litecoin, Monero, and Stratis. Because of the developments in cryptocurrencies blockchain apps are used hugely. [2]

Blockchain is a developing and new distributed consensus system that allows transactions, and any other data storing, to be securely stored and verified without the need for any centralized authority. Blockchain technology is not centralized like other network components, it is a distributed network, which means that it is not controlled by a single entity or central administration, but run by every node who participant in the network. Blockchains, such as Bitcoin, is supported and hosted by thousands of people worldwide. So all data or the ledger of the transaction is not at the understanding of a single company. The great benefit of blockchain technology is being distributed and do not have to trust a single company with communicating parties' data. Instead of putting data in central authority, data is kept at the entire network of thousands of different people who are all acting independently [2].

Blockchains provide data integrity across a large number of transactional nodes by providing all participants in the ecosystem with a working proof of decentralized trust by the assurance of integrity without using a trusted third party to the transaction system. A blockchain system replaces this trusted third party. A blockchain is a cryptographically linked list of blocks created by nodes where each block has a header, the relevant transaction data to be protected, and additional security metadata like sender identity, signature, last block number, and other header information. It enables decentralized consensus by being a distributed ledger which is called a distributed database or list of records, while simultaneously preventing revision or changing of such records altogether. Blockchains resistant to modification of the underlying data are perceived as embodying a tamper-resistant incorruptible decentralized digital ledger for economic or logical transactions related to virtually anything of value. The blockchain provides universal accessibility, incorruptibility, openness, and the ability to store and securely transfer data. Many applications of blockchains have emerged in the recent past beyond the original applications of cryptocurrency, such as bitcoins. The data can represent a wide variety of elements, documents, facts, packets, transactions, agreements, contracts, monetary transactions, or signatures [3].

A blockchain can support a wide range of tasks, including allowing parties to draw up trustworthy contracts, storing sensitive information, and transferring money safely without the intervention of an intermediary.

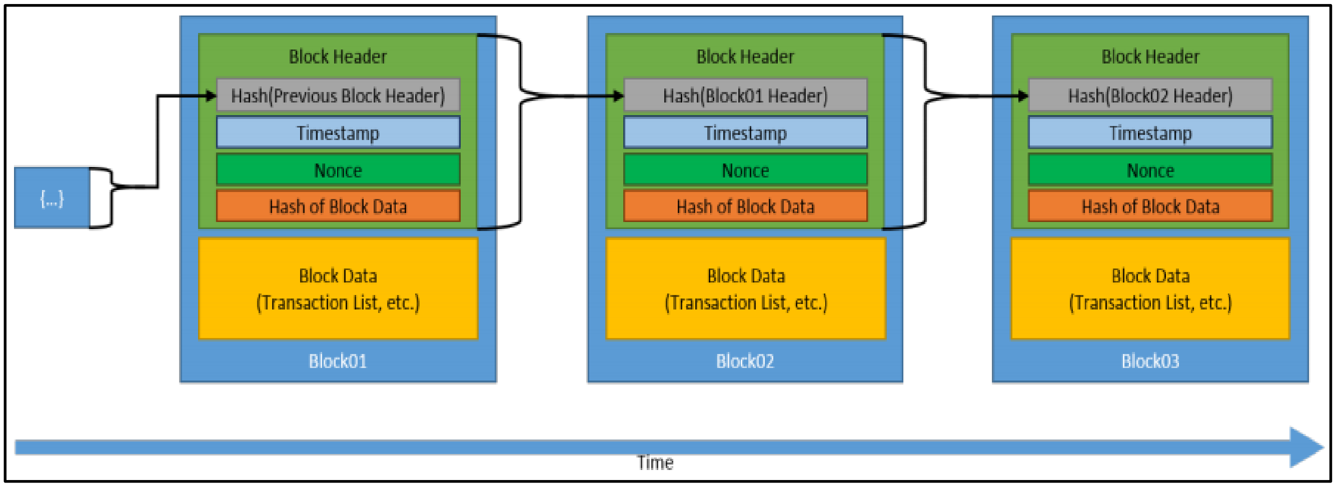
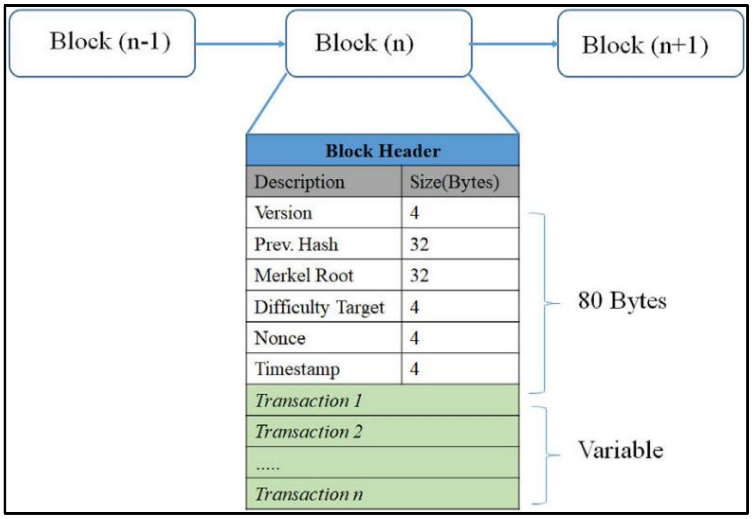


Figure 2-4: Blockchain Transaction [14]

The root of the blockchain is a genesis block that is the first block in the blockchain. It is the common origin of all blocks and contains the information that is generally known to all nodes. The block consists of cryptographic hashes of records, with each block holding the information about the previous block's hash, forming a chain of data, and creating a blockchain. The blockchain begins with a genesis block on top of which stacked the successor blocks. The structure of each block contains a block header and a block body. The block header consists of a previous block's hash, nonce, timestamp, as well as the Merkle root as shown in Figure 5

.



Block Header	
Description	Size(Bytes)
Version	4
Prev. Hash	32
Merkel Root	32
Difficulty Target	4
Nonce	4
Timestamp	4
Transaction 1	
Transaction 2	
.....	
Transaction n	

Figure 2-5:Blockchain Header Format [16]

The block body contains lists of transactions and some additional data, depending on the requirement of the blockchain. Each current block is interconnected with the previous block, using the hash of the previous block like a chain. For immutability, the transactions should be hashed using a Merkle hash which needs to be included in the block header. The Merkle hash is derived from the Merkle algorithm [22], which is a cryptographic algorithm that hashes all transactions of the block to get the Merkel root. The Merkle root is the hash of all hashes of all transactions and it is eventually appended in the block header. The benefit of the Merkel tree is to verify transactions and does not include the body of all transactions in the block header, while still giving a way to validate the whole blockchain. It generates a unique hash value that verifies the integrity of all transactions below it and the size of the Merkel hash is very small as compared with the whole size of all transactions.

The blockchain has gained huge popularity due to its security features of using cryptography. Nowadays, several researchers from different fields, such as academia, financial institutions, medical and banking industries are attracted to the blockchain due to its huge advantages.

2.1.2.1 Blockchain Architecture

The blockchain is a sequence of blocks, which holds a complete list of transaction records like a conventional public ledger (Lee Kuo Chuen, 2015). Figure 6 illustrates an example of a blockchain. Each block points to the immediately previous block via a reference that is essentially a hash value of the previous block called parent block. It is worth noting that uncle blocks (children of the block's ancestors) hashes would also be stored in the Ethereum blockchain (Buterin, 2014). The first block of a blockchain is called the genesis block which has no parent block.

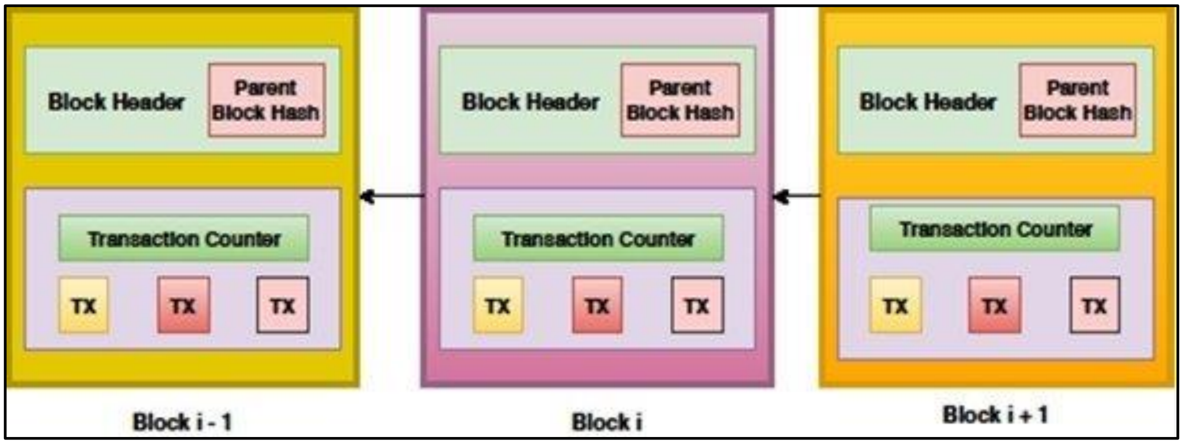


Figure 2-6: : Continuous Sequence of Blocks [7]

Block in blockchain consisting of the block header and the block body. The block body is containing a transaction counter and transactions. The maximum number of transactions that a block can contain depends on the block size and the size of each transaction. Blockchain uses an asymmetric cryptography mechanism to validate the authentication of transactions. The block header includes the following items:

- Block version: indicates which set of block validation rules to follow.
- Parent block hash: a 256-bit hash value that points to the previous block.
- Merkle tree root hash: the hash value of all the transactions in the block.
- Timestamp: current timestamp as seconds since.
- n Bits: current hashing target in a compact format.
- Nonce: a 4-byte field, which usually starts with 0 and increases for every hash Calculation

2.1.2.2 Application Area of Blockchain

The inspiring features of blockchain technology are attractive to multiple organizations for using it for securing assets. By analysing the suitability and importance of blockchains concerning the needs of each organization it is selected by financial institutions, enterprises like POs (post Operator), risk management, land registration, energy-saving, education, voting system, supply chain, and many others.

- ➔ **Finance Services:** The reserve of blockchain systems such as Bitcoin (Nakamoto, 2008) and (hyperledger, 2015) has brought a huge impact on traditional financial and business services. We can use other financial organizations like Asset Management, Insurance Claims, Processing, and Cross-Border Payments.

- ➔ **Enterprise transformation:** the evolution of financial and business services, blockchain can help traditional organizations to complete the enterprise transformation smoothly. Consider an example of postal operators (Pos). Since traditional postal operators (POs) act as a simple intermediary between merchants and customers, blockchain and cryptocurrency technology can help POs to extend their simple roles with the supplying of new financial and non-financial services. Jaag et al. (2016), explored opportunities of arising blockchain technology for POs and requested that each PO could issue their post coin which is a kind of colored coin of Bitcoin.

- ➔ **Risk management:** The risk management framework plays a significant role in financial technology (FinTech) and now it can be combined with blockchain to perform better. Pilkington (2016) provided a novel risk-management framework, in which blockchain technology is used to analyses investment risk in the Luxembourgish scenario. Investors who nowadays hold securities through chains of protectors tend to face the risk of any of these failings. With the help of blockchain, investments and collateral can be decided quickly instead of going through long-term consideration.

- ➔ **Land registration**: One of the typical blockchain applications in the public services area land registration (NRI, 2015). Land information such as the physical status and related rights can be registered and publicized on blockchains. Besides that, any changes made on the land, such as the transfer of land or the establishment of a loan using the land can be recorded and managed on blockchains consequently improve the efficiency of public services.

- ➔ **Energy-saving:** blockchains can be used in green energy. Gogerty and Zitoli (2011) proposed the solar coin to encourage the usage of renewable energies. In particular, a solar coin is a kind of digital currency used for rewarding solar energy producers. In addition to the usual way of getting coins through mining, solar coins could be granted by the solar coin foundation as long as the organization has generated solar energy.

- ➔ **Education:** Blockchain is originally devised to enable currency transactions to be carried out in a trustless environment. But learning and teaching process as the currency, blockchain technology can potentially be applied to the online educational market. In Devine (2015), blockchain learning was proposed. In blockchain learning, blocks could be packed and placed into blockchain by teachers and the learning achievements could be thought of as coins.

- ➔ **Voting system:** we can use blockchain in the voting system because smart contracts on blockchain can be used. Ethereum already built-in smart contracts which can be easily used for a voting system.

- ➔ **Supply chains:** in the supply chain parties involved in the supply chains system, so blockchain provides this system. Blockchain can easily nodes or users in this case join the system and this makes ease the communication among the supply chain. Everything will be visible for all parties at all times, making the whole process run smoother.

2.1.2.3 Characteristics of Blockchain

Blockchain is a decentralized database in which data and information can be stored, and adaptable to deal with the transaction of assets. However, it is based on important pillars that make it different from another database and one of the leading developing technologies [5].

- ▪ **Open distributed ledger**: Blockchain is a decentralized network and a database is distributed, and copies of all information are shared among the participants in the network. All participants can validate this information without the need for a centralized authority. If a transaction is changed, a new block is created and chained to the previous blocks. Record data between nodes of the Blockchain network are matched at random intervals. This randomly matching is makes blockchain technology secure from hackers, as there is no bank information or identities of the parties and the data is public in real-time [5]. The decentralization combined with the real-time updating of information makes Blockchain good in networks involving different organizations.

- **Rules to share data**: all Participants administer the Blockchain. They agreed in advance the types of transactions, which are stored in the chain as smart contracts.
- **Few intermediary third parties:** traditional business transaction involves two parts: a public ledger entry about the transaction and private messages between the parties involved about identities, security keys for transactions and location (Korpela et al., 2017). The combination of these two parts and the decentralization of the system accessible to anyone who validates makes it possible to avoid the intermediary trusted third party (i.e., banks, exchanges, brokerage firms, or price reporting agencies), executing a transaction with limited cost and time, and in a secure way.
- **Consensus-based and trustiness:** due to the decentralized storage nature and the presence of more than one copy of the database, participants have to agree by consensus, on the source of truth and thus validate the transaction. The consensus mechanism allows avoiding that mistakes or fake actions which could affect the database.
- **Cryptographically closed and Immutability of data**: any data records made in a blockchain cannot be changed or deleted. cryptographic technologies are needed for digital signatures and data integrity for avoiding the manipulation of a block, once the transaction has been validated and recorded. This cryptographic mechanism makes data stored in the Blockchain absolute and unique.
- **Time-stamped and Chronological blocks**: Blockchain is composed of a chronologically orderly link of blocks. They let a user create analytics based on dynamic data.
- **Forgery Resistant:** A decentralized solution where the transactions are public is susceptible to different kinds of attacks. Cryptographic hash and digital signatures can be used to ensure the blockchain system is forgery resistant. If a transaction signs a hash of it, no one can alter the transaction later and say you signed a different transaction. Similarly, you cannot later claim you never did the transaction, because it is you who signed it.

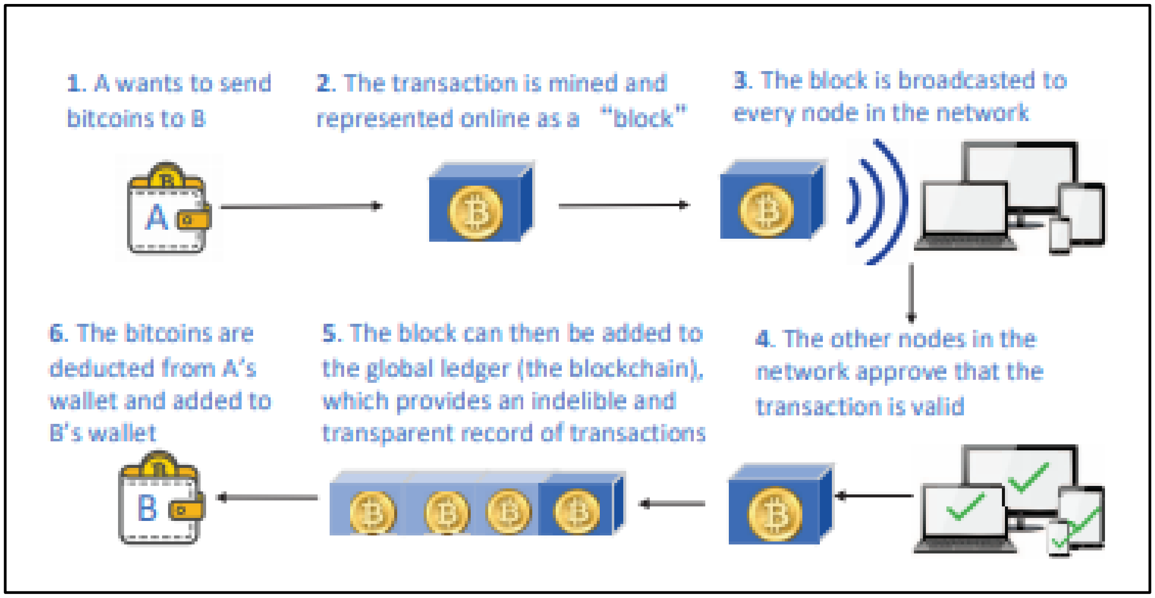

1. A wants to send bitcoins to B
2. The transaction is mined and represented online as a "block"
3. The block is broadcasted to every node in the network
4. The other nodes in the network approve that the transaction is valid
5. The block can then be added to the global ledger (the blockchain), which provides an indelible and transparent record of transactions
6. The bitcoins are deducted from A's wallet and added to B's wallet
A
B

Figure 2-7: How Blockchain Works [15]

2.1.2.4 Types of Blockchain

Blockchain technologies can be divided into three types. Public Blockchain, Consortium Blockchain, and Private Blockchain.

1. **Public Blockchain**

Instead of using a central server public blockchain is secured by cryptographic verification supported by incentives for the miners. Anyone can be a miner to combined and publish the transactions made by the participants. Everyone can check the transaction and verify it, and can also participate in the process of getting consensuses like Bitcoin and Ethereum. This is what we mean by a peer-to-peer (p2p) program. The public blockchain is also a peer-to-peer program with one very important difference is that participants cannot move files (data) from peer to peer, it also ensures that all the peers have the same data. If the data changes on one machine, it changes on all the machines. Rules are specifying exactly how a change can be made, and if someone doesn't follow them and modifies their copy illegally, they're ignored from the network and data communication. It's no different from an email program trying to send an email without the proper SMTP headers it won't be recognized by other email programs. As the above email scenario in public blockchain if the version gets deleted or corrupted, it’s not a problem, just re-sync with peers and can get a fresh valid copy of the transaction [22].

In the current public blockchain working principles like Bitcoin and Ethereum instead of changing data within the data set, new data is just appended on the old data, or data is only written, never deleted. This is how it gets the name blockchain, because new data is added in batches, or blocks, and appended to the existing blocks, forming a chain of blocks. Not only does everyone have the same database (blockchain), but everyone gets a locker within the blockchain that only they can access.

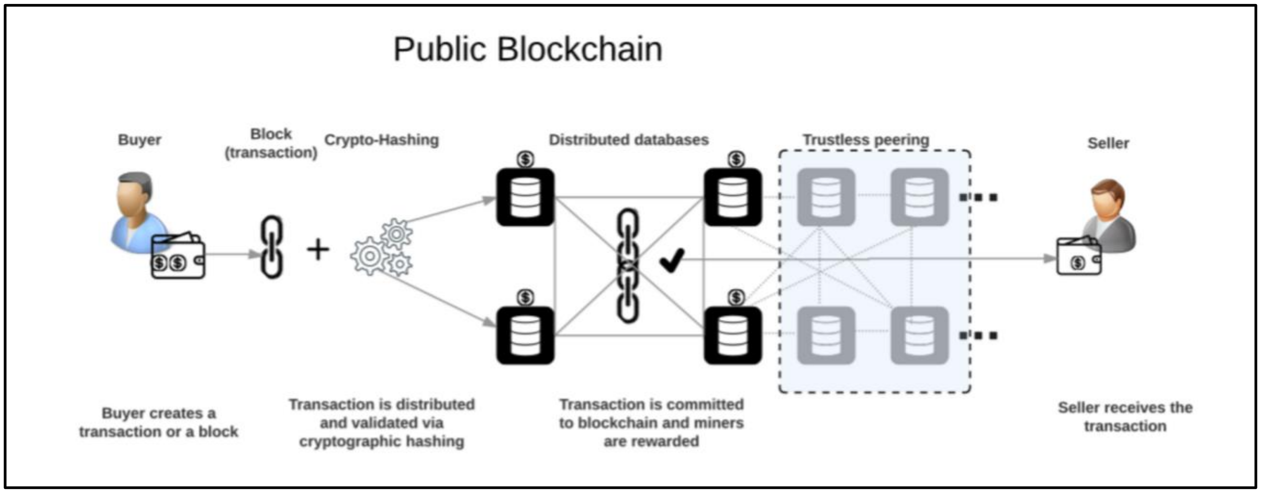

Public Blockchain
Buyer
Block (transaction)
Crypto-Hashing
Distributed databases
Trustless peering
Seller
Buyer creates a transaction or a block
Transaction is distributed and validated via cryptographic hashing
Transaction is committed to blockchain and miners are rewarded
Seller receives the transaction

Figure 2-8: Public Blockchain [22]

2. **Consortium Blockchain**

In Consortium Blockchain the node that had authority can be chosen in advance. The data in Blockchain can be open or private which can be seen as Partly Decentralized like Hyper ledger. The right to read the blockchain may be public or restricted to the participants, and there are also hybrid routes such as the root hashes of the blocks being public together with an API (Application Programming Interface) that allows members of the public to make a limited number of queries and get back cryptographic proofs of some parts of the blockchain state. These sorts of blockchain are distributed ledgers that may be considered partially decentralized [22].

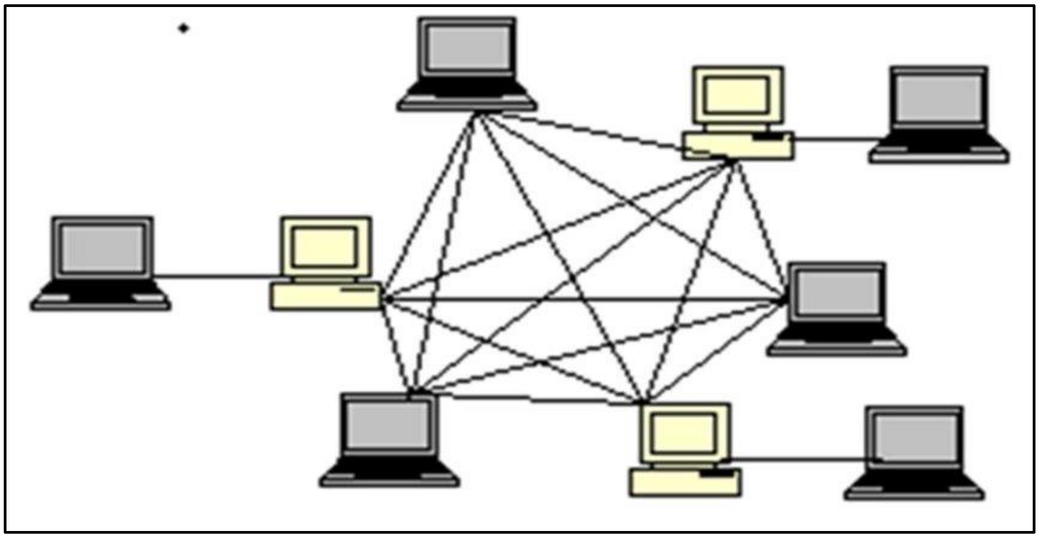


Figure 2-9: Consortium Blockchain [22]

3. **Private Blockchain**

In a private type of blockchain, nodes will be restricted in data access and management. A fully private blockchain is a blockchain where the write permissions are kept centralized to one organization. Read permissions may be public or restricted to an arbitrary level. Like database management and auditing internal data of a single company, public readability may not be necessary in many cases at all, though in other circumstances public audibility is desired. Private blockchain could provide solutions to financial enterprise problems [22].

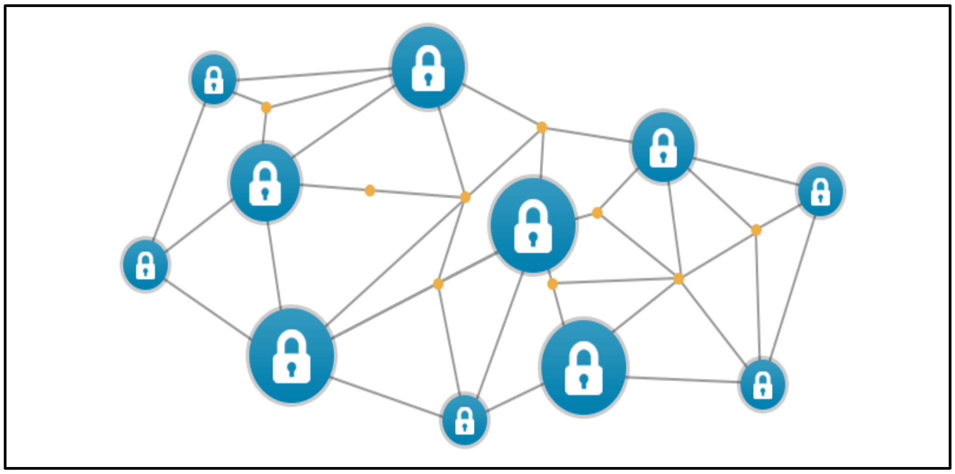


Figure 2-10: Private Blockchain [22]

2.2 Related Work

2.2.1 MANET Security

Many researchers have studied on securing transmission from different types of attacks and publicize their solutions in recent past years [4, 5, 11, 12, 13]. Security attacks of mobile ad-hoc networks are classified into two categories: passive attacks and active attacks. In a passive attack, the malicious node does not disturb the operation of data therefore it is very difficult to detect the node in the network. It includes traffic analysis, monitoring, and eavesdropping. Most Encryption algorithms are used to prevent passive attacks. On the other hand, in an active attack, a malicious node interrupts the normal functioning of the network system by performing either external attacks or internal attacks. External attacks are from malicious nodes that do not belong to the network and External attacks can be prevented by using cryptography techniques. Internal attacks are from either compromised or hijacked nodes that attempt to disturb the normal routing function to consume the network resources. Internal attacks include modification, impersonation, jamming, and denial of service (DoS) attacks which are very difficult to prevent [12]. In this work, we need to address these five major security services to prevent the security attacks of MANET: Availability, Confidentiality, Authentication, Integrity, and Non- repudiation.

The enhancements made to the Bellman-Ford calculation incorporate opportunity from circles in the routing table. Each mobile node in the network keeps up a routing table in which the majority of the achievable goals inside the network and the number of jumps to every route data are recorded. These types of routing algorithms is DSDV which is a table-driven protocol dependent on the traditional Bellman-Ford system. While AODV, DSR, and TORA share the on-request manner to start routing action. AODV utilizes a table-driven routing structure and topology arrangement numbers. DSR uses a source routing and TORA uses a connection inversion routing system. Commonly AODV, DSR, and TORA have a less routing load and DSDV has a less end-to-end delay [29].

AODV (Ad-hoc On-Demand Distance Vector Routing) protocol is a type of MANET routing protocol and many attacks can be implemented to break the connections on AODV based Ad-hoc networks. The aim of securing MANET by modeling the AODV protocol with one type of attack and analyzed the vulnerabilities of the protocol. Based on the analyzing results, the researcher proposed an enhancement to the AODV protocol against the Black Hole Attacks [38].

security problems and energy efficiency are considered as the major factors in MANET. However, the security threats occur due to their limited resource characteristics of MANET. Hence their functionalities are highly degraded with numerous security attacks like Black hole attacks. The black hole attack mainly troubles the data collection and makes an effort to occupy as many links as possible to increase the resource-constrained issues in the network. To solve these issues, the researcher proposes a novel trust-based energy-aware routing (TEAR) mechanism for MANETs. The most important characteristic of the proposed mechanism is it mitigates black hole attacks by a dynamic generation of multiple detection routes to detect the attacker quickly and provides better data route security by obtaining the node trust. TEAR mechanisms are generated by only utilizing the energy in no hotspots without wasting the energy to improve the energy efficiency and desired data route security. The TEAR mechanism highly optimizes the lifespan of the network by avoiding black hole attacks and hugely increasing successful data routing [39].

Due to dynamic topology of MANET, the network encounters frequently broken link that affects the communication. Due to the decentralized nature of the MANET network, the security system poses a higher dimensional challenge for authentication and authorization. One of the complex security problems in MANET is to identify the behavior of nodes. Various cryptographic-based studies have been conducted in the past to provide better security. However, more or less every prior study considers certain attack scenarios and then deploys the design of the attack mitigation model. One of the disadvantages of such a study is that the countermeasures techniques are highly specific and are not applicable when the adversarial scenario changes [18].

Security in mobile ad-hoc networks (MANETs) is a key issue to be addressed. There are two separate approaches for security in MANETs which are continuous authentication and intrusion detection system. In continuous authentication work, these two classes of security approaches are integrated and combined into a single console. Intrusion detection systems (IDSs), serving as the second wall of protection, can solve this problem and effectively help identify malicious activities. An IDS continuously or periodically monitors the current matter activities, compares them with stored normal profiles or attack signatures, and initiates proper responses. Authentication is an important type of response initiated by an IDS. Authentication is the process of verifying the identity of a user. Depending on the network elements and the authenticators, there could be different kinds of Authentication mechanisms. The user can be associated with confidential information which he or she is supposed in possession of such as a password, a private key, a distinctive physical or logical address, a fingerprint, a retina scan, and a voice or speech pattern. After an authentication process, only authentic users can continue using the network resources and negotiated users will be excluded [35].

The ad-hoc networks are vulnerable to different attacks that range from eavesdropping to active interference due to all communications being performed over the air. Any kind of mechanism, system, protocol, or network is also vulnerable to a large number of attacks and malfunctioning. The systems are based on pure trust and that is exactly what the malicious users interfere with the network and device [36].

Another vulnerability is application security, network security, information security. Attacks to the wireless ad-hoc network within the networking layer sometimes have two purposes the first one is not forwarding packets or adding and changing some parameters of routing messages, sequence change, and information processing addresses. Using cryptography or authentication mechanisms can prevent attackers. However, these mechanisms defend the network against attacks that return from outside attacks [37].

In the MANET network if there is a malicious node it can do much damage to the network. Most damages are discussed in the above paragraph. When an attack is appearing in the network the result may be an end-to-end delay, denial of services, packet loss, or modification of packets. In this proposed work nodes are authenticated node to node and end-to-end. So this authentication is a crucial point of data integrity, confidentiality, and non-repudiation. In MANETs network if an attacker is in the network the network performance and security of the network is decreased in many aspects. To protect the network and the data from this situation Blockchain Cryptographic -based Authentication and Verification for Secure Communication in Mobile Ad-hoc Network (MANET) by modifying AODV protocol is proposed. Mobile Ad-hoc Networks where the basic concept of authentication is modified from the existing symmetric key-based authentication perform better in insider attackers [17].

In symmetric key-based authentication [17] mechanism has proposed to ensure secure communication between the communicating MANET Nodes. The proposed model secures the network from the well-known and frequently occurred attacks (impersonation, modifies routing information, black hole). In this work, two levels of authentication have been used, first level is hop-to-hop authentication (MD5 algorithm has used for authentication code generation) and the second level for end-to-end authentication (SHA1 algorithm has for authentication code generation). Digest_1 is the size of 128 bits generated by the MD5 algorithm and Digest_2 is the size of 160 bits generated by the SHA 1 algorithm.

The main limitation of this work is that their result shows out of all malicious nodes 50% of nodes are taken as inside attackers and the remaining 50% as the outside attacker. Their simulation results also show that their model performs better in the presence of an inside attacker. So we need to address the outside attack.

Our proposed work is more secure in terms of ignorance of any malicious node and rejecting the transmitted message of blocks and put the node in the rejected list. The performance of the proposed algorithm is evaluated in two ways. The first one is Network performance metrics which are attack detection rate (DR), False Negative Rate (FNR), and false Positive Rate (FPR). The second one is in terms of network performance metrics: packet delivery ratio (PDR), the average end-to-end delay (AE2ED), and network throughput (TH). The proposed algorithm has better in data transmission and its rejection of malicious nodes is better. Moreover, it also restricts the ability of the malicious nodes to cause further damage to the network by refusing the message and putting it in the rejected list to tell other nodes. In this work, we want to analyze the security challenges of MANET with the proposed solution.

2.2.2 Blockchain Security

A decentralized personal data management system that guarantees user's ownership and control over his/her data by using blockchain technology. Blockchain technology is used to develop a protocol for automated access control while not requiring central third-party management or administration. In the growth research industry concerning the Internet of things(IoT) and how blockchain technology is used to provide security and privacy in the peer-to-peer networks with topologies such as IoT [27].

In the Vehicular Ad-hoc Networks (VANET), the collection and broadcasting of life-threatening traffic event information by vehicles are important. However, the traditional VANETs face several security issues. In [30] paper the researchers propose a new type of blockchain to resolve critical message broadcasting issues in the VANET. They create a local blockchain for real-world event message exchange among vehicles within the boundary of a country, which is a new type of blockchain suitable for the VANET. They propose using a public blockchain that stores the node trustworthiness and message trustworthiness in a distributed ledger that is appropriate for secure message dissemination. And also they do not adopt existing blockchain directly applicable for VANET scenarios. They propose a new type of blockchain which is suitable for the VANET and use event messages as transactions in the VANET. The researcher believes that the blockchain can resolve the major issues faced by current VANETs and provide security for critical information broadcasting [30].

Understanding blockchain technology for future supply chains is to influence future practices and policies in the system. The paper offers valuable insight for supply chain experts into how blockchain technology has the potential to disrupt existing supply chain provisions as well as several challenges to its successful diffusion. The study is one of the first studies to examine the current state of blockchain diffusion within supply chains. It lays a firm foundation for future research [25].

Because of current blockchain requires high calculation capability and it has overhead and delays that are not suitable for IoT devices, it is not suitable for IoT networks. In [33] research work proposed an architecture that is a lightweight blockchain for IoT that removes the overhead of the existing blockchain while maintaining security and privacy as a solution. The proposed IoT architecture uses a blockchain but has a centrally managed node to maximize battery efficiency. Central nodes with sufficient computing power and storage space compared to sensor nodes create overlay networks to implement public blockchain that ensures end-to-end security and privacy. The proposed architecture uses distributed trusts to reduce block verification processing time. The smart home was used as a test target to test the efficiency of the proposed architecture. The simulation results showed that the proposed architecture reduced packet and processing overhead compared to the existing blockchain. The proposed architecture addresses the low computing power and low storage issues of the IoT device as blockchain, but the existing security issues are defective because there is a central node that integrates data considering battery issues of the sensor device [33].

The unique feature of blockchain technology is its reliable and suitable services for data security. It also discusses the security issues and challenges behind blockchain technology. Blockchains are seen as highly scalable technologies that can be applied in many areas. One of them is that it is a key technology that can complement weaknesses in areas where security is weak [28].

Applying blockchain to IoT systems has been carried out. IoT has improved its suitability and usefulness to more areas beyond its scope of application. The research work proposed to overcome the limitations of resources in lightweight networks such as IoT and implementing public ledger for lightweight networks. The proposed protocol uses one of the popular social networks Twitter and blockchains to ensure transactions secure manner. By sending and receiving messages between users, the process of sharing information with many users is verified. Blockchains can be built by making and connecting blocks with verified messages. The social network-based blockchain is ideal for lightweight networks than the traditional blockchain because it does not require high computing power and more storage capacity [34].

Blockchain technology uses many techniques to achieve the security of transaction of data or block of data, irrespective of the user of data in the block. Many applications such as bitcoin use encryption technology for data safety. The other most secure concept of blockchain is that the longest chain is the authentication process. This reduces the security risks up to 51% majority attack and the possibility of data loss problems. As the longest chain is the ultimately authenticated attacks become null and void as they end up being left [26].

When blockchain technology used in VANET, Vehicles obtain their Pseudo IDs from the Certificate Authority (CA), which are stored along with their certificate in the immutable authentication blockchain and the pointer corresponding to the entry in blockchain and enables the Road Side Units (RSUs) to verify the identity of a vehicle on road. This framework identifies the malicious node and shares relocated vehicle with their identity in the shared blockchain network [10].

Bitcoin uses a technology called Blockchain, which was founded on its peer-to-peer nature and lack of a trusted central authority. Cybersecurity of vehicular ad-hoc networks (VANET) has become a prime research area for cybersecurity researchers. Minor vulnerabilities in VANET may lead to a severe loss of life. This problem has engaged many researchers with the purpose of improving the defense mechanism of these susceptible networks. The paper [30] presents a brief meta-analysis of the field of cybersecurity of autonomous vehicles using Blockchain and discusses how VANETs and CAVs can be protected using the same blockchain technology.

VANET runs on the principles of the mobile ad-hoc network (MANET) that enables automatic data exchange between vehicles through a wireless network. The beginning of Bitcoin became a sensation in the cyber world. The main reason behind its popularity and success is based on its security measures. It offers benefits such as integrity, security, and privacy. Blockchain uses a distributed algorithm to operate, although it contains remarkable concord among its blocks. The paper scholars say they contribute to the improvement of VANET's overall performance using Blockchain [31].

Because high-performance security algorithms are used in securing data it is difficult to apply in a device with limited resources, there is a problem that security is low in lightweight networks such as IoT. Internet of Things (IoT) is a lightweight network consisting of sensor devices that can be connected to the Internet and can communicate wirelessly. Scholars proposed a method that guarantees stability when updating IoT devices by using blockchain in IoT networks. The proposed method uses the non-variable characteristics of the blockchain. If an update is added to the blockchain with a valid block, the attacker is not possible to erase the block. The application of the blockchain can be used to verify that these updates are secure after a consensus process between IoT devices. IoT blockchain for the security of updating IoT devices is more effective when the size of the IoT network grows. Because many devices only download reliable updates that have been validated, they can increase the lifespan of the device by preventing unnecessary resource waste [32].

2.2.3 Summery

Table 2-1: Summary of Surveyed Literature Review

No	Year	Authors	Publisher	Title	Purpose	Simulation tool	Conclusion	Future work
1	2017	D. Ram Kumar C. Annadurai K. and Nirmala Devi	Springer	Continuous authentication console in MANET	verify the presence of the authentic user continuously during the lifetime of the MANET network	NS-2	Using the four-level of security decrease the intrusion and decrease the probability of intrusion on the system	Multiple biometrics authentication need more memory to overcome this issue
2	2018	Sachin Malhotra and Munesh C. Trivedi	Springer	Symmetric based authentication mechanism for secure communication in MANET	Protect the network from impersonation, modification of route information, and black hole attack	NS-2	Better in the insider attacker,	Only identify insider attacker the outside attacker
3	2018	Nisha Malik Priyadarsi Nanda, Arushi Arora, Xiangjian He and Deepak Puthal1	IEEE	Block-chain based security identity authentication and revocation framework for VANET	Reduce the competition and communication overhead by mitigating dependency on a trusted authority for identity verification	Omnet ++	Eliminate the need to circulate CRLs by the CA or RSUs and disseminate the status of a relocated vehicle	decentralize the VANET environment and aim to use smart contracts particularly with an emergency scenario
4	2018	Sandeep A. Thorat, P. J. Kulkarni and S. V. Yadav	Springer	Verification and validation of trust-based opportunistic routing protocol	Analytically proof the function of the protocol and test using a t-test to verify and validate the statistical result	Analytical proof and SPSS 20.0	Loop freedom and packet delivery ratio is better	to verify the working of other newly proposed routing protocols by using automated model checking tools like SPIN and HOL.

5	2019	Fu, Y., Li, G., Mohammed, A., Yan, Z., Cao, J., and Li, H	IEEE	A Study and Enhancement to the Security of MANET AODV Protocol against Black Hole Attacks	Enhancing AODV protocol against the Black Hole Attacks	NS-3	the proposed methods can be used to protect the MANET from Black Hole Attacks in some degrees	adopt machine learning method into the proposed method to improve the efficiency of the proposed methods. And study Worm Attacks and DDoS attacks
6	2019	Merlin, R. T., and Ravi, R.	Springer	Novel Trust-Based Energy-Aware Routing Mechanism for Mitigation of Black Hole Attacks in MANET	Increase scalability, security, and increase probability of successful routing	NS -2	The proposed mechanism highly optimizes the lifespan of a network by avoiding black hole attacks and drastically increasing the probability of successful data routing	evaluate dynamic blacklists by setting different values to certain variables
7	2020	Ningthoujam Chidananda Singh, D., and Sharma, A	ResearchGate	Understanding the MANET Security Using Various Algorithms and Types	Discuss working and the efficiency of the defense mechanism that is being used to fight the security issues in the network		effective and efficient and is capable of curing the problems related to security in MANET	improvement to chip away at these exemptions and improve the effectiveness of CBDS conspire
8	2020	A Gupta, N Ranjan	IJSRET	A Survey of Attacker Identification and Security Schemes in MANET	consider different attacks and security schemes and proposed a new approach against wormhole attack in MANET	NS-2	security scheme against attack is also mention that improves the network performance.	use profile base detection technique and we prevent black hole attack using a neighbor trustworthy base technique

Chapter Three

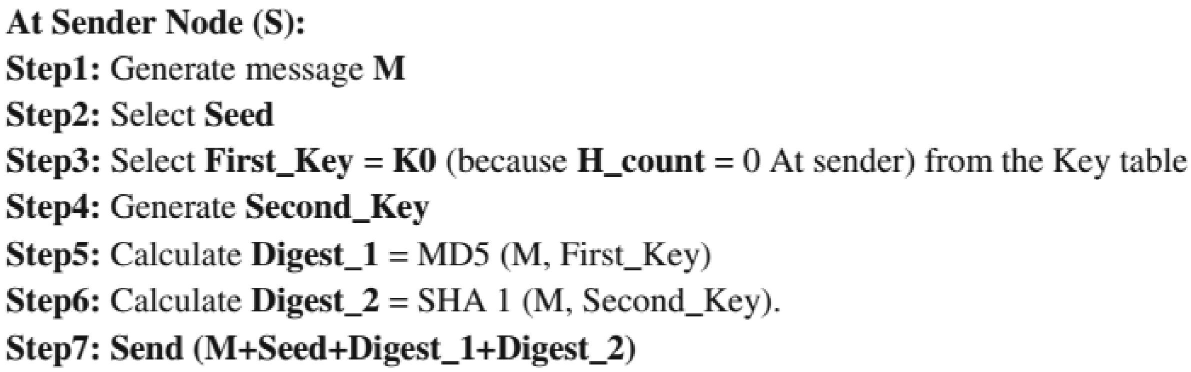
Proposed Block-Chain Cryptographic -based Authentication and Verification for Secure Communication in MANET

In this chapter we discuss about detail of why we choose Blockchain technology, authentication, verification Responsibility of Nodes to Perform the Proposed Method on Network, Explanation of Proposed Model, and description of proposed algorithm.

Blockchain authentication and verification for secure communication in MANET focus on securing the communication among MANET nodes and improving the network performance and security performance of MANET. Our research work considers modifying a reactive routing protocol which is called Ad-hoc On-Demand Distance Vector (AODV) and implemented within the NS-2 simulator. The security mechanism is designed for the Mobile ad-hoc networks with various scenarios using the NS-2 simulation tool in the AODV routing protocol.

The modified base paper symmetric key authentication algorithm is performed at the sender node, intermediate node and receiver node. The main modification is focused in adding blockchain security for authentication and verification of node and message.

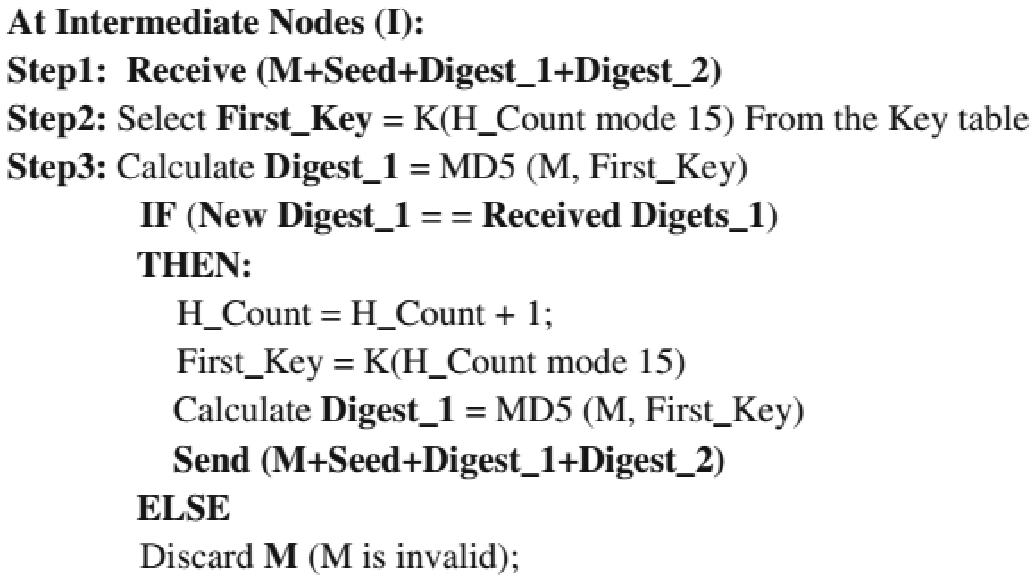
- The modified algorithm of symmetric key authentication algorithm at the sender node



```
At Sender Node (S):
Step1: Generate message M
Step2: Select Seed
Step3: Select First_Key = K0 (because H_count = 0 At sender) from the Key table
Step4: Generate Second_Key
Step5: Calculate Digest_1 = MD5 (M, First_Key)
Step6: Calculate Digest_2 = SHA 1 (M, Second_Key).
Step7: Send (M+Seed+Digest_1+Digest_2)
```

Figure 3-1: Modified Algorithm at the Sender Node [17]

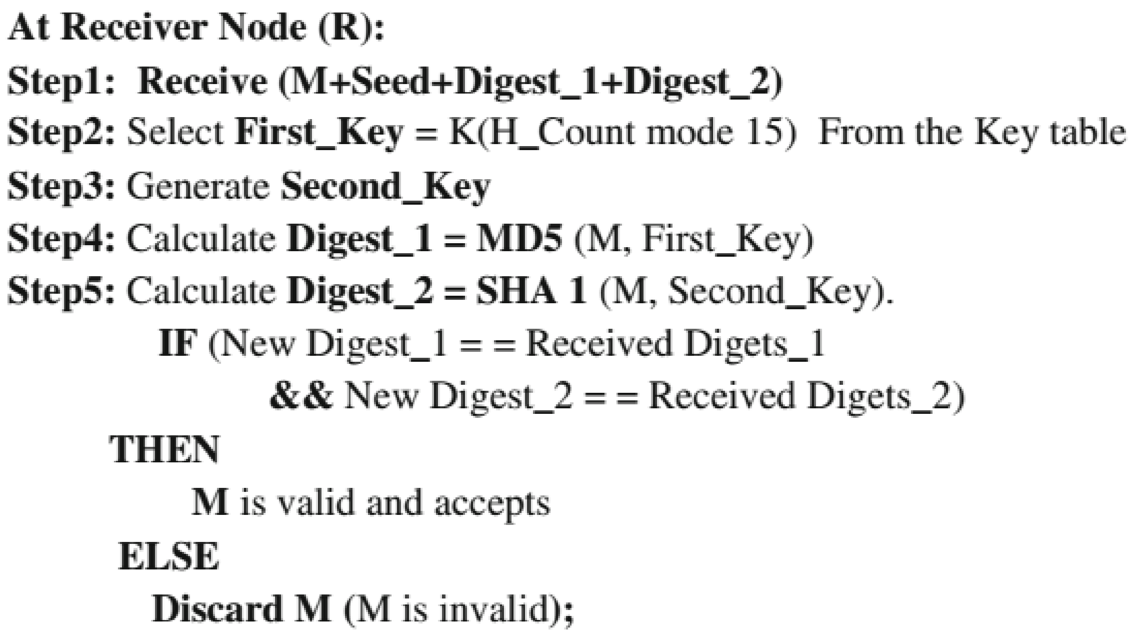
- The modified algorithm of symmetric key authentication algorithm at the intermediate node



```
At Intermediate Nodes (I):
Step1:  Receive (M+Seed+Digest_1+Digest_2)
Step2: Select First_Key = K(H_Count mode 15) From the Key table
Step3: Calculate Digest_1 = MD5 (M, First_Key)
           IF (New Digest_1 = = Received Digets_1)
           THEN:
              H_Count = H_Count + 1;
              First_Key = K(H_Count mode 15)
              Calculate Digest_1 = MD5 (M, First_Key)
              Send (M+Seed+Digest_1+Digest_2)
           ELSE
           Discard M (M is invalid);
```

Figure 3-1: Modification at the Intermediate Node [17]

- The modified algorithm of symmetric key authentication algorithm at the receiver node



```
At Receiver Node (R):
Step1:  Receive (M+Seed+Digest_1+Digest_2)
Step2: Select First_Key = K(H_Count mode 15)  From the Key table
Step3: Generate Second_Key
Step4: Calculate Digest_1 = MD5 (M, First_Key)
Step5: Calculate Digest_2 = SHA 1 (M, Second_Key).
            IF (New Digest_1 = = Received Digets_1
                    && New Digest_2 = = Received Digets_2)
        THEN
              M is valid and accepts
        ELSE
           Discard M (M is invalid);
```

Figure 3-2: Modification at Receiver Node [17]

3.1 Why Blockchain?

There is no doubt that blockchain is a hot issue in recent years, although it has some topics we need to notice, some problems have already been improved along with new techniques developing on the application side, getting more and more mature and stable.

The blockchain is highly appraised and endorsed for its decentralized infrastructure and peer-to-peer network nature. Blockchain could be applied to a variety of fields far beyond Bitcoin its application areas are discussed in chapter two. Blockchain has shown its potential for transforming the traditional industry with its key characteristics: decentralization, persistency, privacy, and audibility [3].

Blockchain could be well combined with big data. We roughly categorized the combination of blockchain technology into two types data management and data analytics. Data management blockchain could be used to store important data in s distributed and secure state. Blockchain could also ensure the data is original. For example, if blockchain is used to store patient’s health information, the information could not be altered and it is hard to steal that private information. The data analytics category transactions on blockchain could be used for big data analytics. For example, user trading patterns might be extracted [25]. Users can predict their potential partners’ trading behaviors with the analysis.

In the blockchain, the communications are secured using cryptography. Using cryptography ensured that a valid node in the network is initiating the communication and no one can forge a false communication. This means, cryptographically it can be ensured that the sender node can make the communication so no one can duplicate (make forged) the sender signature and also the intermediate nodes signature.

3.1.1 Authentication

Authentication must be the main part of MANET data sharing to secure the relations between every node in the network. We use the symmetric cryptography technique for authentication of the node. Symmetric cryptosystems are usually faster and more useful when the data size is huge. We select SHA-1 because its 160-bit hash functions consumed 512-bit block sizes. SHA-1 was used in the digital signature algorithm (DSA). It was used quite a lot in many security tools and Internet protocols such as SSL, SSH, TSL, etc.

Reasons why we select the hash function is:

- Hash functions are used in verifying the integrity and authenticity of the information.
- Hash functions also are used to index data in hash tables. This can speed up the process of searching. Instead of the whole data, if we search based on the hashes instead of compared with the whole data then it should be faster. That is the main advantage of using hash functions in the proposed MANET security.
- Node securely authenticate without storing the passwords locally
- Hash functions are one-way functions

Among all key management mechanisms for MANET, we use the pre-distribution key mechanism. We use a key table of K0- K15 for generating the MD5 algorithm which generates 128 bits digest for checking the integrity of the block of a message at each node in the network during the route request process. The pre-distribution mechanism provides a suitable balance between storage load and processing ability.

3.1.2 Verification

Like authentication, verification is an essential part of MANET data sharing. The verification process takes place by checking the correctness of authenticated node and data message it comes from the sender node. The verification process is performed at the intermediate and destination node.

We use the Merkle tree for the verification of nodes in the MANET network. A Merkle tree is a binary tree of cryptographic hash pointers. Merkle trees are constructed by hashing paired data then again hashing the hashed outputs up to the root node, which is named Merkle root, and constructed bottom-up. If in case an attacker needs to altering data at any level in the tree it would not match with the hash stored at one level up in the hierarchy, and also till the root node. It is difficult for an attacker to change all the hashes in the entire tree. It also ensures the integrity of the order of communications of nodes.

To verify the incoming data, the node starts to calculate the hash of the sender node and intermediate node and, and see if it matches the sender node hash. Then continue checking the hash up to the data is received by the destination node. Continuing this process to the top root hash is the quickest possible way for transaction verification (just Log (n) time for n items). In this case, the parent hash is the sender node hash.

3.1. 3 MD5 (Message Digest 5)

R. Rivest of RSA Data Security Inc. has designed a series of hash functions, that were termed MD for “message digest” followed by a number from 1. A cryptographic hash function works by mapping data to a fixed length string of characters. These types of hash functions are used in many ways. They can be used for authentication, indexing data into hashed tables, checksums, and digital signatures [42]. We select the most used cryptographic hash functions for authentication MD5 which is uses a 128-bits hash value. MD5 works by taking variable length data and converting it into a fixed length hash string of 128-bits [41].

3.1.4 SHA-1 (Secure Hash Algorithm 1)

NIST (National Institute for Standards and Technology) proposed Secure Hash Standard (SHS) that contains the description of the Secure Hash Algorithm (SHA). This hash function is designed to work with the Digital Signature Algorithm (DSA) proposed in the Digital Signature Standard (DSS). SHA is based on the hash function MD4, and its design closely models MD4. SHA-1 produces a hash value of 160 bits [41] Secure hash algorithms (SHA) accept a message of predefined size, then through several steps of compression function calculations, the output hash is produced.

3.1.5 Markel Tree Algorithm

A Merkle tree is a binary tree of cryptographic hash pointers, hence it is a binary hash tree. It is another useful data structure used in blockchain. Merkle trees are created by hashing paired data at each node then again hashing the hashed outputs all the way up to the root node, which is called the Merkle root. Like any other tree, it is constructed bottom-up. Similar to the hash pointer data structure, the Merkle tree is also tamper-proof. Altering at any level of tree would not match with the hash stored at one level up in the hierarchy, and also root node. It is really difficult for an attacker to change all the hashes in the entire tree. It also ensures the integrity of the order of transactions. If any change in the order of the transactions, then also the hashes in the tree till the Merkle root will change. Merkle trees provide efficient way to verify if a specific data communication belongs to a particular block. If there are “n” data communication in a Merkle tree, then this verification takes just Log (n) time [43].

3.1.6 Responsibility of Nodes to Perform the Proposed Method on Network

Blockchain authentication and verification security method is appropriate only when multiple parties share their data, which might later have changed and would need a view of final common information. However, multiple parties sharing data is not the only qualifying criteria for blockchain authentication and verification to be a possible solution in the ad-hoc network. To better the effectiveness of a blockchain authentication and verification security method the node in the network must fulfill the following criteria:

- ✓ Multiple nodes in the network update data when needed: If the request has to record actions and update incoming data from multiple nodes in the network.
- ✓ The requirement for verification: When trust among parties plays a critical role and they should realize that their actions are being recorded as valid
- ✓ Use the Merkle tree algorithm for the verification process

- ✓ Interactions are time-sensitive: When a node is positive for the data sharing it reduces delay and accelerates a transaction.
- ✓ Data exchange interact: When transactions created by multiple participants interact and depend on each other.

3. 2 Proposed Model

The proposed method Blockchain Cryptographic -based Authentication and Verification for Secure Communication in Mobile Ad-hoc Network (MANET) are described in this section. Blockchain Cryptographic-based Authentication and Verification is a work which is securing a mobile ad-hoc network by verification of the node and the messages and assure the messages basic networking security requirements.

Due to the decentralized nature of MANET malicious nodes are entered into the network at any time and make different types of attacks. We will try to secure the network and make efficient communication techniques to the MANET network. When a new node joins the network it must authenticate and verified by its neighbor and other nodes in the network. If the node is legal it joins the network and participates in the network if it is not legal the node itself and the message are rejected from the network and the node is put in the rejected list. After the verification process if the node is legal its block of message is added to the blockchain and the node starts communication. This makes a remarkable difference and efficient results for rejection of malicious nodes and their data from the network and putting it in the rejected list.

In the proposed method, we modify and add some improvements in the Symmetric Key Based Authentication Mechanism for Secure Communication in MANETs [17]. In the proposed algorithm two levels of authentication are used to test integrity, non-repudiation, and confidentiality of the message. The first level of authentication is used in hop-to-hop which is performed at the neighbor or intermediate nodes and the second level of authentication is used end-to-end or at the receiver nodes which is the receiver checks the whole blocks of the chain with the Merkle tree algorithm. All nodes in the network have the responsibility to verify the incoming block of messages. The first level authentication MD5 algorithm is used to generate 128 bits digest for checking the integrity of the block of a message at each level of the node in the network during the route request and message data transferring process. The second level of authentication uses SHA1 to generate the 160 bits digest for checking the integrity of the message and the sender node which is come from the sender node at the receiver node.

Message and node authentication is used to verify the integrity of a message and node. Message authentication assures that the data received are exactly as sent without modification, insertion, or deletion received data message. In many cases, there is a requirement that the authentication mechanism assures that the requested identity of the sender is valid. When a hash function is used to provide message authentication, the hash function value is often referred to as a message digest.

Any new nodes, when it joins the network it develops its message it also generates key1 and key2 then makes the block of all these messages then broadcast its block message. Each node in the network itself is proven and digitally signed to ensure its legitimacy. Before this block is added to the network, it should be verified by the neighbor nodes in the network. The verification and authentication are used to identify the node's status which is the node is malicious or normal node. Our algorithm uses a key table size of 15 keys (Key0 to Key14) which is stored at each node at the time of network deployment. These keys are used for generating MD5 authentication at the first level that helps us to check the integrity of the message and the sender node at the intermediate nodes. For generating the second-level code, which is selected from a random number, and SHA1 is used to generate the second key from the random number generator. The random number generator function uses a four-digit number as a seed to generate the second key. After the two keys, key1 and key2 have generated the node makes the block of the message, key1, and key2. Then it broadcast the block to the Niebuhr node. The intermediate node receives the block of a message and then verifies the incoming block of the messages and the node by generating MD5 authentication.

The node verifies the incoming data by calculating the hash of the sender node and intermediate node and, and see if it matches the sender node hash. If the hash is equal to the sender and intermediate node is considers as a genuine node and accepts its data. And the node appends its key1 to the block and makes the chain of the block and it broadcast the blockchain. In the verification step if the hash is not equals to sender hash the node is considered as malicious node and discarded its message and the intermediate node ignores the node and puts the node ID in the rejected list. Finally, the destination node generates MD5 and SHA1 to authenticate the incoming message and node. To verify it uses Markel tree and checks all intermediate nodes and sender node are genuine. If all calculated hash is equal it accepts the message. But if there is a mismatching hash from all the intermediate nodes and also the sender node isolates the node. Then the destination node rejects its message and puts it in the rejected list.

The following figure shown us how the proposed model works in a MANET network.

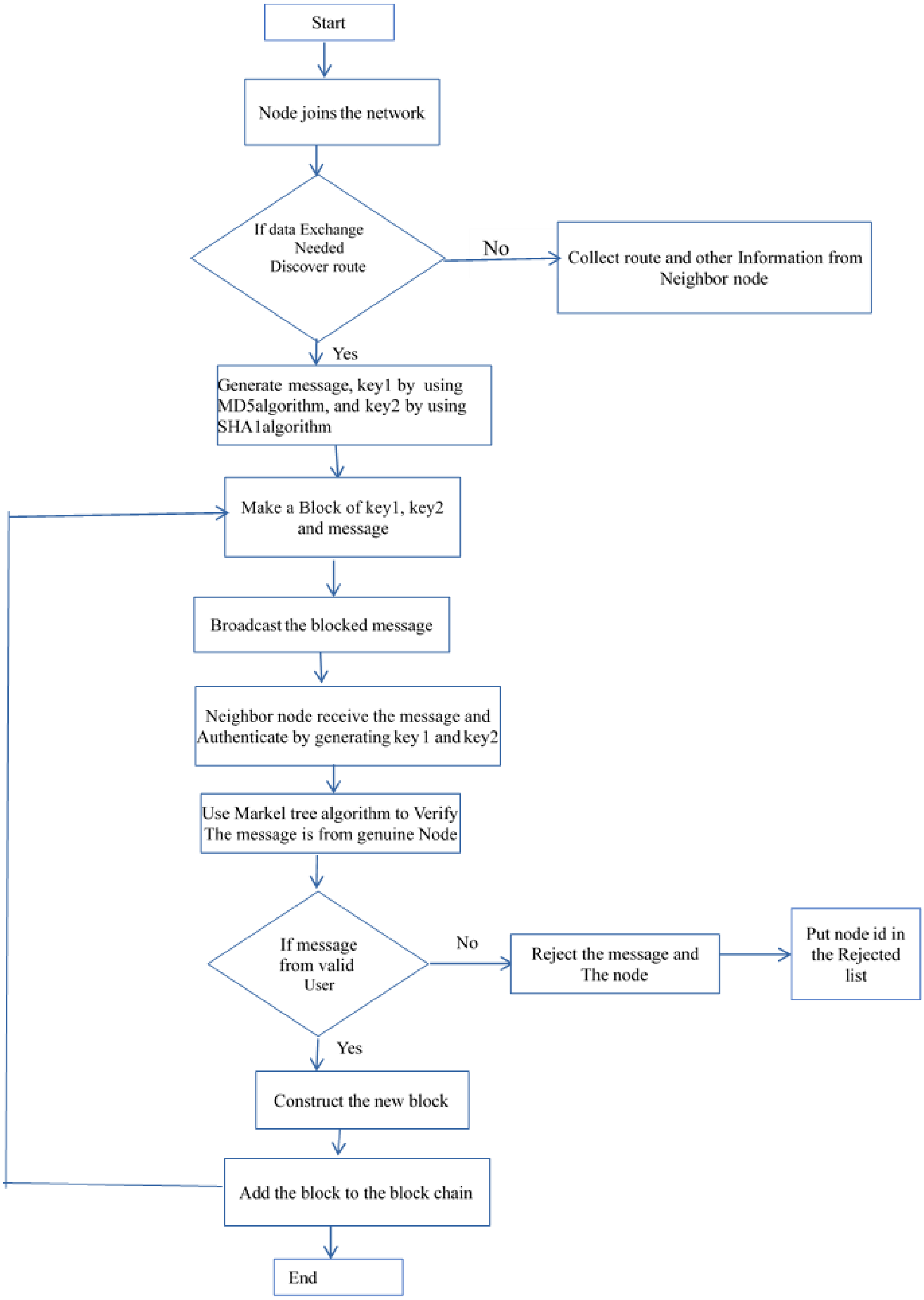

Start
Node joins the network
If data Exchange Needed Discover route
No
Collect route and other Information from Neighbor node
Yes
Generate message, key1 by using MD5algorithm, and key2 by using SHA1algorithm
Make a Block of key1, key2 and message
Broadcast the blocked message
Neighbor node receive the message and Authenticate by generating key 1 and key2
Use Markel tree algorithm to Verify The message is from genuine Node
If message from valid User
No
Reject the message and The node
Put node id in the Rejected list
Yes
Construct the new block
Add the block to the block chain
End

Figure 3-3: Flow Chart of Proposed Algorithm

3.3 Proposed Algorithm

Here, the proposed algorithm is discussed as follows:

The sender node first creates its message, then selects the first key from the key table for generating the Digest1 using MD5. (First key = Key number (Hop count mode 15). To calculate first key, the node use inputs key number from key table according to node hope count value and use MD5 algorithm then use random number generator function to generate the Second key for creating the second level authentication code using the SHA1 algorithm. Then make the block message by combining the message, random 4-digit seed, Digest1(key1) and Digest2 (key2) then make a block of message data to send to the next node in the network. Then the intermediate nodes first authenticate the node and verify the message by generating only the Digest 1 of the message using the key (The first key = Key number (Hop count mode 15) and if new Digest1 equals to the received Digest1, then the message is treated as from valid node. Then the intermediate node again calculates its Digest1 from the key table and the created digest 1 is appended into the block of the message and forwarded to the next node in the network. If new Digest1, does not equal to the Received Digest1 then the message is discarded and the node is rejected from the network and the node puts the node in the rejected list by using its ID. In the intermediate node, the verification process is done by using the Markel tree algorithm. Merkle trees are performed by hashing paired data then again hashing the hashed outputs up to the sender node. Then check the paired hash if it is from a genuine node or not.

Finally, the receiver node generates new Digest 1, and Digest2 is created by using the First key (First key = Key number (Hop count mode 15) from the key table and Second key generating by random number generator using same seed that has been used by the sender). If both digest1 and digest2 new digest equal, authentication is finished and verify the message by using Markel tree algorithm with all received hash values or digest, if the message is received as valid message and the node is genuine. If the digest is not equal the message is discarded as the invalid message and the node is rejected from the network and put in the rejected list.

Proposed Algorithm Requirement:

A) Source node
B) Destination node
C) Intermediary node

D) Key table

E) Seed (four-digit number)

Algorithm Abbreviations

a. HC- hope count

b. M – message

c. K – key

d. Digest 1 - 128-bit message authentic code generated by the MD5 algorithm

e. Digest 2 - 160-bit message authentic code generated by SHA 1 algorithm

f. First key - a key that is selected from the key table used by MD 5

g. Second key - a key that is generated by random number and used by the SHA 1 algorithm

h. Block - it contains all data from the sender, or intermediate node (digest 1, digest 2, and the message)

i. Block-chain- contains number of block in chronological order

j. Seed - 4-digit random number

The algorithm at the sender node

```
1. Start
2. Generate message        // the message which is send to the receiver node
3. Select the first key    // from the key table according to its hop count value
4. Generate key 1          // use MD5 algorithm to generate key 1
5. Generate (calculate) digest 1 == md5 (m, first key)
6. Select seed             // use random number generator function for selecting seed value
7. Calculate digest 2 == SHA1 (m, second key) // for second level authentication
8. Make a Block of M, digest1, digest2        //
9. Send the block to the neighbor node        //
```

The algorithm at the intermediate node

```
1. Receive the block
2. Select the first key from a key table
3. Calculate digest 1 == MD5 (M, first key)
if (calculated digest 1 == received digest 1)
then
```

```
      HC= HC+1
      first key =key (HC mode 15)
      calculate digest 1 = MD5 (M, first key)
  4.  Use the Markel Tree algorithm to check all pair of hashes    // for the verification process
      if (all calculated pair of hash ==received pair of hash)
         then verify the node and the message
  5.  Select another key from the key table // the key is selected according to hope count values
  6.  Generate (calculate) digest 1 = MD5 (M, key)
  7.  Make a block of the message (received message, current digest1)
  8.  Append the block to the previous block to make the block-chain
  9.  Send the block-chain
      else
  10. Discard the message and ignore the node        // in the verification process if one bit of hash value
      is not equals to from the sending blocks of the chain
  11. Put the node ID in the rejected list
```

The algorithm at the receiver node

```
  1.  Receive block-chain
  2.  Select blocks which contain its message from the block-chain
  3.  Select the first key from a key table for authenticating the node
  4.  Generate the first key             // from the key table according to its hop count value
  5.  Calculate digest 1 MD5 (M, first key)
  6.  Calculate digest 2 SHA 1 (M, second key)
      if (generated digest 1 ==received digest 1 and digest 2)
      node is valid
  7.  Use the Markel Tree algorithm to check all pair of hashes to verify the message and the node
      if (all calculated pair of hash ==received pair of hash) the node and message are verified and
      accepted // all intermediate node between the sender and the destination node by using log (n) nodes
else
discard the message and put the Nod ID in the rejected list
End
```

Chapter Four

Simulation Design and Performance Metrics of Block-Chain Cryptographic -based Authentication and Verification for Secure Communication in MANET

To carry out the proposed system, we have to implement it on an appropriate platform and programming language. So, we selected the implementation software tools and implement it in a way that is suitable for the simulation scenario. We have implemented and simulated the designed algorithm in AODV routing protocol using NS-2.35 simulator running on Core i5 @ 2.53GHz CPU, 4.00 GB RAM, and 64-bit with Ubuntu 18.4 operating system. The proposed system implementation and the result of the proposed work are discussed in this chapter in detail. Before that, we want to talk about a short discussion on the network simulator, related software with NS-2, and the implementation of AODV in NS -2.

4.1. Network Simulator (NS-2) Overview

There are many network simulators with different features and platforms. We use the most popular network simulation named NS-2, which is one of the most popular, used in many network scenarios and open-source network simulators. Network Simulator (Version 2) [25], widely known as NS-2, is simply an event-driven simulation tool that has proved useful in studying the dynamic nature of communication networks. Simulation of wired as well as wireless network functions and protocols routing algorithms, TCP and UDP can be done using NS2. In general, NS2 provides users with a way of specifying such network protocols and simulating their corresponding behaviors. Due to its flexibility and modular nature, NS-2 has gained constant popularity in the networking research community.

NS-2 is a discrete-event simulator, where actions are associated with events rather than time. An event consists of execution time, a set of actions, and a reference to the next event. These events connect and form a chain of events on the simulation timeline. Unlike a time-driven simulator, in an event-driven simulator, the time between a pair of events does not need to be constant [25].

NS-2 consists of two key languages: C++ and Object-oriented Tool Command Language (OTcl). The C++ programming language defines the internal mechanism or the backend of the simulation objects. And the OTcl language sets up simulation by assembling and configuring the objects as well as scheduling discrete events or the frontend. NS-2 uses OTcl to create and configure a network.

The C++ language uses to run simulation scenarios. All C++ codes need to be compiled and linked to create an executable file. Since the body of NS-2 is fairly large, the compilation time is not small and it is fast to run but slow to change [25].

On the other hand, OTcl is an interpreter, not a compiler. Any change in a OTcl file does not need compilation. But the OTcl part does not convert all the codes into machine language, each line needs more execution time. OTcl is slow to run but fast to change. It is suitable to run a small simulation over several repetitions with different parameters. NS2 is constructed by combining the advantages of these two languages [25].

NS-2 provides users with an executable command ns which takes on an input argument, the name of a Tcl simulation scripting file. In most cases, a simulation trace file is created and is used to design graphs and create animation. To interpret these results graphically and interactively we use different tools such as NAM (Network AniMator) and XGraph. To analyze a particular behavior of the network, users can extract a relevant subset of text-based data and transform it into a more potential presentation [20].

We can use NS-2 for the following tasks:

- To evaluate the performance of existing network protocols.
- To evaluate new network protocols before use.
- To run large-scale research which is not possible in the real world.
- To simulate various types of networks

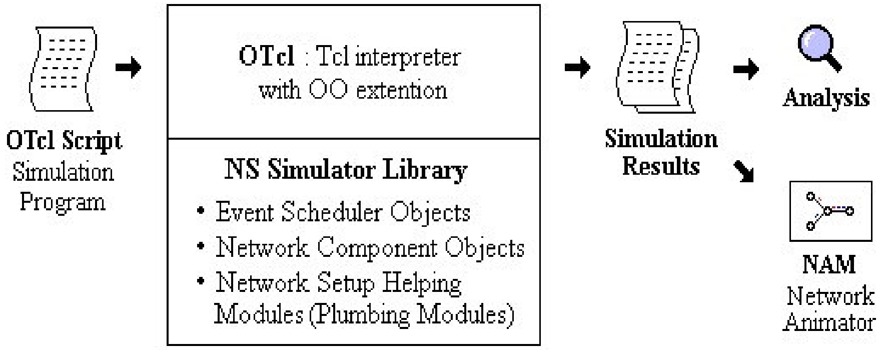


Figure 4-1: NS -2 Simulation Flow[20]

To use NS-2, a user programs in the OTcl script language. An OTcl script will do Initiates an event scheduler, sets up the network topology using the network objects, tells traffic sources when to start and stop transmitting packets through the event scheduler.

Tcl (Tool Command Language) is used by millions of research areas in the world. It is a language with a very simple syntax and allows very easy integration with other languages. The characteristics of this language are:

- ✓ allows a fast development
- ✓ provides a graphic interface
- ✓ is compatible with many platforms
- ✓ is flexible for integration
- ✓ is easy to use and we can get it free from the internet or it is open source

4.1.1 Software Tools and Application Used with NS-2

A. Network Animator (NAM),

When we use NS-2, it has a software tool called network animator (NAM). NAM provides a visual interpretation of the network topology we created [20]. Its features are listed as follows:

- ✓ It provides a visual interpretation of the network which is created
- ✓ It helps to executed animation directly from a Tcl script
- ✓ It has a window that controls commands like play, stop forward, reward, pause, a display speed controller, and a packet monitor facility.
- ✓ It shows information such as throughput, number of packets on each link.

- ✓ NAM provide a drag and drop interface for creating network typologies.

B. Ns-Script

Ns-script is developed in Java. It is a graphical user interface (GUI) used for building NS-TCL scripts. The network topology can be built by simply drawing it in the edit screen. The Nodes & agents can be added by using the simple drag and drop function in the edit screen. Once we create the network topology, the TCL code for the topologies is automatically generated in the TCL script screen. The Ns script is used to create different types of network topologies by simply adding nodes, parameters, and their respective links. Transport agents like UDP, TCP can be created and added to the topology and the simulation events can be scheduled. The scripts created using Ns-script can be exported and executed in NS-2.

C. X-Graph

The X-Graph is used for beneficial plotting and generating graphs. To use X-Graph in NS-2, it should be called within a TCL Script. It will load a graph showing the visual information of the trace file produced in the simulation.

D. Trace Graph

Trace graph is a good application that comes very easily to NS-2 users. It removes the coding of AWK scripts that are needed to configure and run over the trace file. Trace graph makes analysis very simple. Trace graph seems to have been developed using MATLAB and therefore supporting codes are needed to make it run in Linux.

E. NSG 2.1 (NS-2 Scenarios Generator 2 version 1)

NS-2 Scenarios Generator 2 (NSG2) supports and provides some graphical user interface (GUI) wired and wireless scenario by creating nodes, node parameters, and node links using the drag and drop option. Some features of NSG2 are listed below:

- ✓ Provide simplex and duplex links for wired network, grid, random and chain topologies, TCP, and UDP agents.
- ✓ Supports TCP Tahoe, TCP Reno, TCP New-Reno, and TCP Vegas congestion control and avoidance mechanisms.

- ✓ Supports Ad-hoc routing protocols such as DSDV, AODV, DSR, and TORA and also applications like FTP (file transport protocol) and CBR (constant bit rate).

The main limitations of the NSG2.1 are

- ✓ support a limited of protocol,
- ✓ do not distinguish between topology present in wired network types like Ring, Star, Mesh, and other types of wired network
- ✓ Do not distinguish between models present in the wireless scenario, steps for generating scenarios are not clear and cannot provide any kind of help for understanding the GUI system [38].

F. AWK (**Aho, Weinberger, and Kernighan Scripts**)

AWK is a scripting language used for manipulating data and generating reports. The AWK command-based programming language requires no compiling and allows the user to use variables, numeric functions, string functions, and logical operators.

AWK is a utility that enables a programmer to write little codes but effective programs in the form of statements that define text patterns that are to be searched for in each line of a document and the action that is to be taken when a match is found within a line. AWK is mostly used for designing, scanning, and processing. It searches one or more files to see if they contain lines that match with the specified patterns and then perform the associated actions.

The main Operation of AWK is: -

- ✓ Scans a file line by line
- ✓ Splits each input line into fields
- ✓ compares input line and fields to pattern and
- ✓ Acts on matched lines.

4.1.2 Reasons why We Select NS-2 Simulator

Because of its simplicity and modularity, the NS-2 network simulator has gained vast popularity among the research community. It allows simulation scripts, also called simulation scenarios, to be easily written in a like different script programming language and OTcl. The functionality of NS-2 depends on C++ code that either comes with the NS-2 program itself or is supplied by the user. This flexibility makes it easy to enhance the simulation environment as the need of the user. Even though most common parts are already built-in. such as wired nodes, mobile nodes, links, queues, agents (protocols), and applications. Most network components can be configured in detail, and models for traffic patterns and errors can be applied to a simulation to increase its reality.

The most advantages of NS-2 are listed below:

- ✓ Ns-2 is cheap, it does not require costly equipment
- ✓ It used to simulate wireless networks and wired networks
- ✓ It supports many MANET protocols
- ✓ Support Traffic Source Behavior like www, CBR, VBR
- ✓ Support Transport Agents (UDP and TCP)
- ✓ Used to create Network Topology easily
- ✓ It has Applications like Telnet, FTP, and Ping
- ✓ Used to trace Packets on all links or specific links
- ✓ Complex scenarios can be easily tested.
- ✓ Results can be quickly obtained more ideas can be tested in a smaller time frame.
- ✓ Supported platforms: it supports multiple platforms (Windows, Linux, Mac, and others)
- ✓ Modularity to measure the overall structure of networks
- ✓ It is Popular and most networking related researches are done with it

4.2 AODV Routing Protocol and Reason Why We Select AODV Protocol

Ad-hoc On-Demand Distance Vector Routing (AODV) [28] is a reactive routing protocol that creates a destination path when it needs. The routes are not built until certain nodes intend to communicate or transmit data with each other. AODV has better performance than other MANET routing protocols [29]. It is also the most discussed, compared, and extended protocol by many researchers.

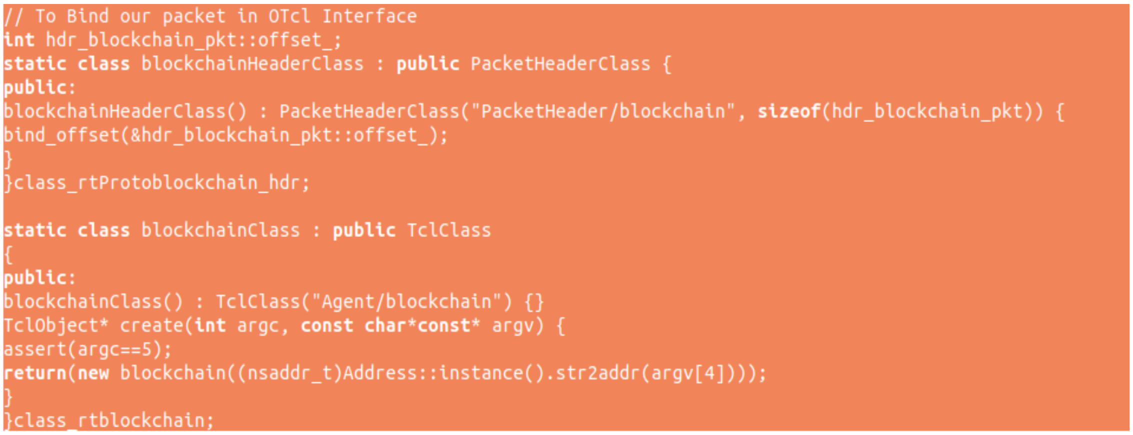
The AODV protocol keeps a routing table to store the next-hop routing information for destination nodes. Each routing table can be used for a period of time. If a route is not requested within that period, it expires and a new route needs to be found when needed. Each time a route is used, its lifetime is updated. When a source node has a packet to be sent to a given destination node, it looks for a route in its route table. If there is one route in its table, it uses it to transmit the packet. Else it initiates a route discovery process to find a route by broadcasting a route request (RREQ) message to its neighbor nodes [30].

AODV stores only one routing entry per destination. So using AODV routing protocol have the benefit of decrease memory overhead, minimum network resources use, and better in high mobility situation. It does not support multi-path routing. Though this reduces the overhead at each node, it creates a disadvantage, especially during a route failure event. When an active link is broken, AODV has to initiate a new route discovery process which would incur additional delay and network flooding [32].

The advantage of the reactive schemes is that they do not consume a large amount of network bandwidth and resources. From other reactive protocols we choose AODV because of the following reasons:

1. AODV consumes less memory, compared to other reactive routing protocols, which consumes more memory for a route cache.
2. AODV is efficient in high mobility networks but other protocols are efficient only in networks with less or no mobility.
3. Source packet size in other protocols is large due to route cache. AODV uses less packet size compared to other protocols.
4. AODV can respond to topological change very quickly from other protocols
5. AODV supports unicast and multicast packet transmission
6. Lower delay setup for connection and detection of the latest route to the destination
7. It does not put additional overhead on the data packet
8. It is loop-free, self-starting, and support a large number of a mobile node
9. It does not need any central administration for handling the routing process

In the implementation process, we modify all AODV components aodv.h, aodv.cc, aodv_rtable, aodv_packet.h, and aodv_rtable.h and other files which are uses to trace a file and make the change in AODV protocol.



```
// To Bind our packet in OTcl Interface
int hdr_blockchain_pkt::offset_;
static class blockchainHeaderClass : public PacketHeaderClass {
public:
blockchainHeaderClass() : PacketHeaderClass("PacketHeader/blockchain", sizeof(hdr_blockchain_pkt)) {
bind_offset(&hdr_blockchain_pkt::offset_);
}
}class_rtProtoblockchain_hdr;

static class blockchainClass : public TclClass
{
public:
blockchainClass() : TclClass("Agent/blockchain") {}
TclObject* create(int argc, const char*const* argv) {
assert(argc==5);
return(new blockchain((nsaddr_t)Address::instance().str2addr(argv[4])));
}
}class_rtblockchain;
```

Figure 4-2: Sample Blockchain Modification

4 .3 Implementation Performance Metrics

We have identified security performance metrics and Network performance metrics to compare the base paper [17] to the proposed Blockchain Authentication and Verification for MANET method. Under security performance metrics we select Attack Detection Rate, False Positive Rate, and False Negative Rate, and also we select Throughput, Average end to end delay and, packet Delivery Ratio to evaluate the Network performance of the proposed and the existing system.

4.3.1 Security Performance Metrics

4.3.1.1 Attack Detection Rate (DR)

Attack Detection Rate is defined as the ratio of malicious node correctly identified as a malicious node to a total number of the existing malicious node.

$$Attack\ Detection\ Rate = \frac{(Number\ of\ Detected\ Malicious\ Node\ )}{(Total\ number\ of\ Malicious\ node)}\ \ X100\%$$

4.3.1.2 False Negative Rate (FNR)

False Negative Rate is the ratio of malicious node incorrectly identified as a genuine node from the total number of genuine nodes in the network.

$$False\ Negative\ Rate = \frac{(Number\ of\ Malicious\ Node\ detected\ as\ Normal\ Node\ )}{(total\ Number\ of\ Normal\ node)}\ \ X100\%$$

4.3.1.3 False Positive Rate (FPR)

False Positive Rate is the ratio of genuine node incorrectly identified as a malicious node to the total number of normal nodes in the network.

$$False\text{-}positive\ Rate = \frac{(Number\ of\ normal\ Node\ detected\ as\ Maliciuous\ node\ )}{(total\ Number\ of\ malicious\ node)}\ \ X100\%$$

4.3.2 Network Performance Metrics

Network performance is often defined by the service quality of the network. And we can evaluate it by an average end-to-end delay, throughput, and packet delivery ratio.

4.3.2.1 Average End-to-End Delay

It is defined as the time taken for a data packet to be transmitted across a MANET from source to destination [29].

$$\text{End To End Delay} = \frac{ReceivingTime - SentTime}{No.PacketReceived}\ seconds$$

4.3.2.2 Packet Delivery Ratio (Fraction)

It is calculated by dividing the number of packets received by destination by the number of packets originated from the source. [29]

$$\text{Packet Delivery Ratio} = \frac{PacketReceived}{PacketSent}\%$$

4.3.2.3 Throughput

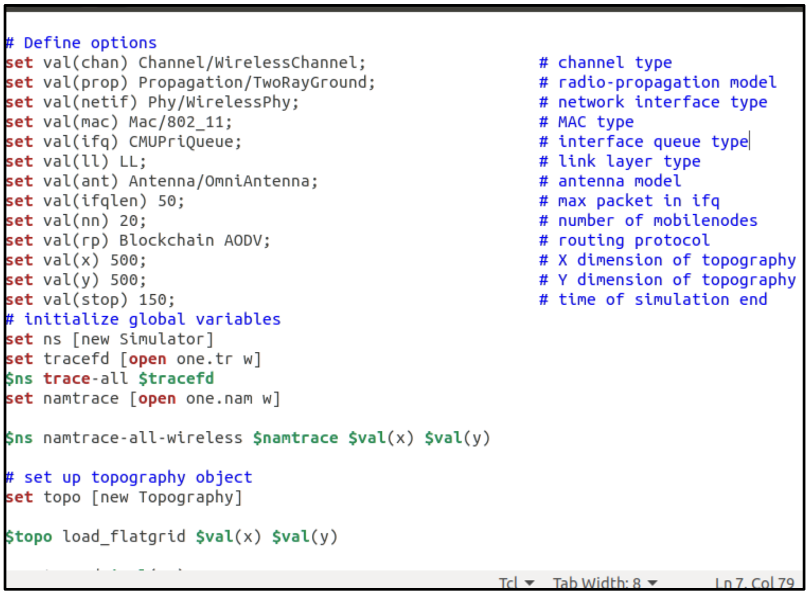
It is the ratio of the total number of bits transmitted throughout a given transmission time. The difference between data transmission end time (Tend) and starts to time (Tstart). This metric depicts how the congestion control mechanism at the source node is affected by the packet loss caused by malicious nodes. A decrease in throughput is the outcome of a malicious node [29].

Throughput = $\frac{bitstransmitted}{EndTime-StartTime}bps$

4 .4 Simulation Scenario Design

To study the feasibility of the proposed system, network simulator 2 (NS2) is used to conduct series of experiments to evaluate its effectiveness in securing MANETs and improving its performance. The process of creating Mobile Nodes consists of Mobile Node Configuration and Mobile Node Construction steps. Each node is assumed to be equipped with a wireless transceiver operating on 802.11 wireless standards. The physical radio frequency characteristics of each wireless transceiver are with a bit rate of 2Mb/sec and a transmission range of 250 meters. Moreover, Each Mobile Node has a buffer that can hold up to 50 packets. The service discipline is based on a prioritized queue which gives priority to routing packets. The type of antenna used is an Omani-directional antenna.

The Random Waypoint Model is used as a mobility model. A node in this model randomly chooses a destination point (waypoint) in the area and moves with constant speed on a straight line to this point. After waiting a 10s pause time, it chooses a new destination and speed, moves with constant speed to this destination, and so on. The destination points are uniformly randomly distributed on the system area (500 by 500m). For generating traffic scenarios, Constant Bit Rate (CBR) traffic sources are used for generating data packets. The source and destination pairs are spread randomly over the network.



```
# Define options
set val(chan) Channel/WirelessChannel;                  # channel type
set val(prop) Propagation/TwoRayGround;                 # radio-propagation model
set val(netif) Phy/WirelessPhy;                         # network interface type
set val(mac) Mac/802_11;                                # MAC type
set val(ifq) CMUPriQueue;                               # interface queue type
set val(ll) LL;                                         # link layer type
set val(ant) Antenna/OmniAntenna;                       # antenna model
set val(ifqlen) 50;                                     # max packet in ifq
set val(nn) 20;                                         # number of mobilenodes
set val(rp) Blockchain AODV;                            # routing protocol
set val(x) 500;                                         # X dimension of topography
set val(y) 500;                                         # Y dimension of topography
set val(stop) 150;                                      # time of simulation end
# initialize global variables
set ns [new Simulator]
set tracefd [open one.tr w]
$ns trace-all $tracefd
set namtrace [open one.nam w]

$ns namtrace-all-wireless $namtrace $val(x) $val(y)

# set up topography object
set topo [new Topography]

$topo load_flatgrid $val(x) $val(y)
```

Figure 4-3: Sample Tcl Script For Implementation

For this simulation study, 20 up to 50 nodes were randomly distributed in a simulation area of 500mx500m. The simulation is executed for 150 seconds of simulation time. The pause time used is 10s and a maximum speed of 10m/s is used for all cases. The traffic connections are constant bit rate (CBR) used and with a flow rate of 4 packets/s and a packet size of 512B.

Table 3-1: Simulation Parameters

No	Parameters	Value
1	Simulator	Network simulator 2 (NS-2)
2	Simulation area	500m x500m
3	Transmission range	250m
4	Number of simulation nodes	20-50
5	Simulation time	150s
6	Mobility speed	10 m/s
7	Pause time	10s – 100s
8	Packets Rate	4 pkts/s
9	Packet size	512 bytes
10	MAC type	802.11
11	Traffic type	CBR (constant bit rate)
12	Transport agent	UDP (User datagram protocol)

We use 500mX500m simulation area because we use maximum of 50 nodes in the network the existing system symmetric key authentication uses 800m × 600m with number maximum of 150 nodes. Oure transmission range is 250M to minimize the number of hops or intermediate node between the sender and destination nodes.

Random Wayward mobility model is the most commonly used mobility model in research community especially Mobile Ad hoc network. Pause time is a time which is specified for once a mobile node start moving in a simulation area before moving to the next destination node. At every instant, a node randomly chooses a destination and moves towards its destination with a speed chosen randomly from a uniform distribution for every mobile node in the network. After reaching the initial destination, the node stops for a duration defined by the pause time parameter. After this pause, it again chooses a random next destination and repeats the whole process until the simulation ends.

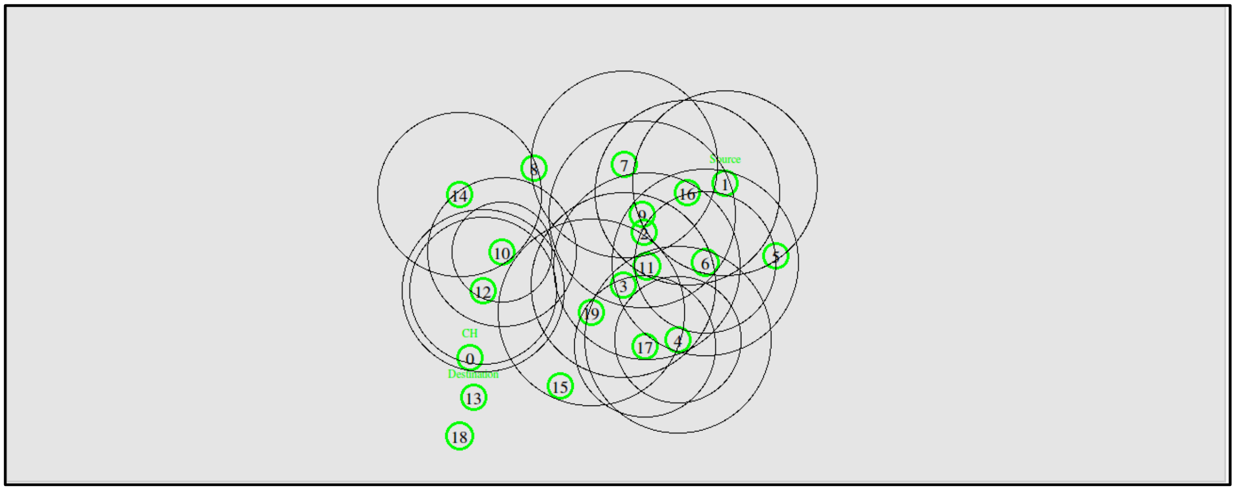
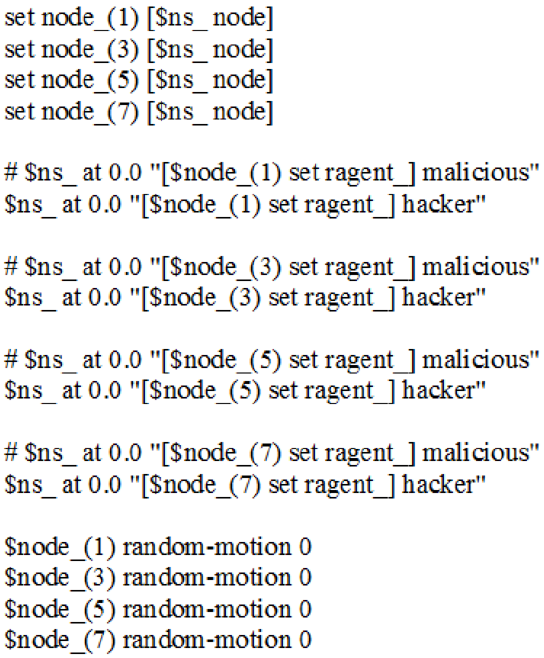


Figure -4: Ns-2 Simulation Of The Implementation

In Figure 4:4 shows the sample network architecture that is used to test the proposed system without attack using summation parameter. For this sample scenario we used the number of nodes randomly 20 nodes. Generally, the main architecture of the network for the experiments is placing the nodes random and makes them mobile and with mobility speed 10 m/s there are multiple connection in the network. The nodes are move randomly with their diameter with variable pause time.

For the purpose of the evaluating performance of our algorithm we add malicious node in the network. We use AODVE malicious node pseudo code for the adding malicious node in the network.



```
set node_(1) [$ns_ node]
set node_(3) [$ns_ node]
set node_(5) [$ns_ node]
set node_(7) [$ns_ node]

# $ns_ at 0.0 "[$node_(1) set ragent_] malicious"
$ns_ at 0.0 "[$node_(1) set ragent_] hacker"

# $ns_ at 0.0 "[$node_(3) set ragent_] malicious"
$ns_ at 0.0 "[$node_(3) set ragent_] hacker"

# $ns_ at 0.0 "[$node_(5) set ragent_] malicious"
$ns_ at 0.0 "[$node_(5) set ragent_] hacker"

# $ns_ at 0.0 "[$node_(7) set ragent_] malicious"
$ns_ at 0.0 "[$node_(7) set ragent_] hacker"

$node_(1) random-motion 0
$node_(3) random-motion 0
$node_(5) random-motion 0
$node_(7) random-motion 0
```

Figure 4-5: pseudo code for malicious node

Chapter Five

Simulation Result Performance Evaluation and Discussion

5.1 Security Performance of Proposed Algorithm

The security performance of the proposed method is evaluated based on Attack Detection rate (DR), False Negative Rate (FNR), and False Positive Rate (FPR).

5.1.1 Detection Rate (DR)

Detection rate is a ratio of the number of malicious nodes detected from the total number of the existing malicious nodes in the network. As a detection rate Table 3 shows, the analysis of the malicious node detection rate scenario that was observed in the existence of variable 1, 2, 3, and 4 malicious nodes which are presented in the total of 50 node scenario in the network. As shown in Figure 5:3 the attack Detection rate of the proposed system is relatively better than the existing system. As the detection rate shows the average detection rate (ADR) of our security method is higher than the existing system, which indicates our proposed algorithm is secured.

Table 4-1: Detection Rate Table

Number of a mobile node	**Number of a malicious node in the network**	**Detected by Symmetric Key Authentication mechanism (Existing system)**	**Detected by blockchain cryptographic authentication and verification (Proposed system)**
50	1	1	1
	2	1	2
	3	1	2
	4	2	3

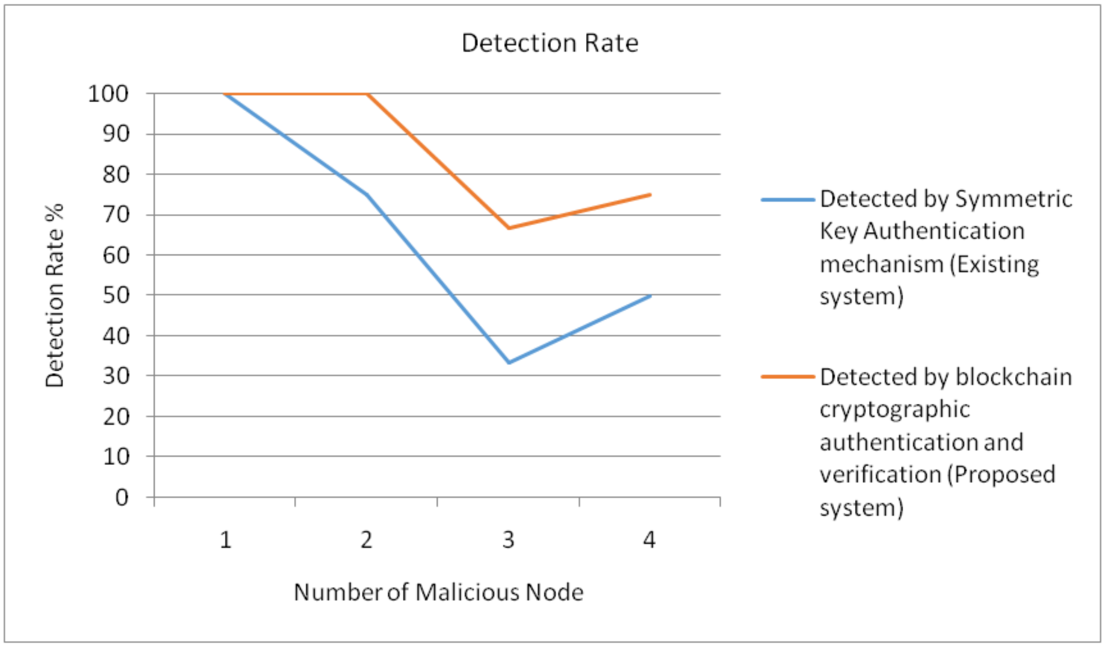


Figure 5-1: Attack Detection Rate with Number of malicious node in the network

5.1.2 False Negative Rate (FNR)

False Negative Rate is the ratio of malicious node incorrectly identified as a genuine node from the total number of malicious nodes in the network. In the existing system, the rate of False negative is higher compared to our proposed system and our proposed system is lower in detected malicious node as a normal node.

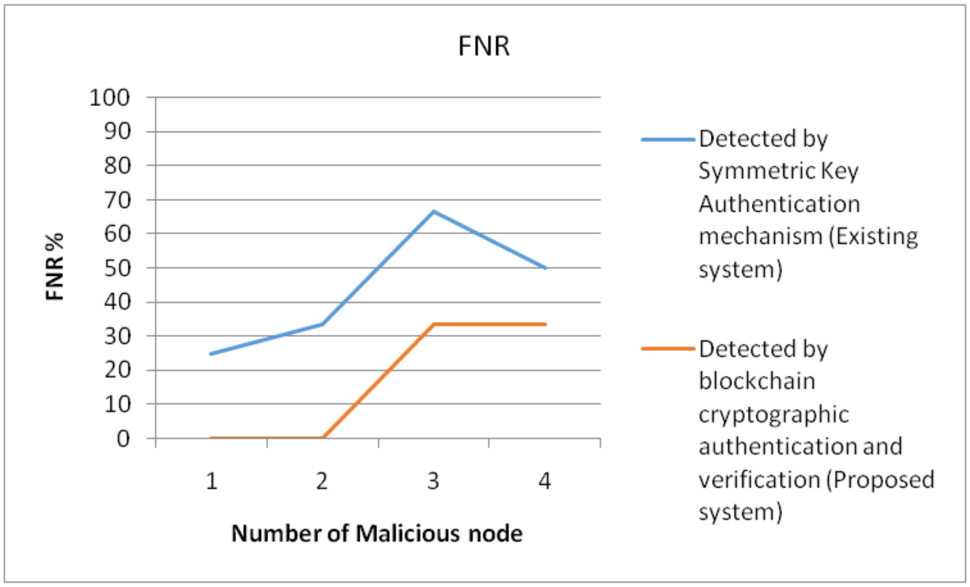


Figure 5-2: False Negative rate of proposed method

5.1.3 False Positive Rate (FPR)

False Positive Rate is the ratio of genuine node incorrectly identified as a malicious node to the total number of normal nodes in the network. A false positive or false alarm is an alert caused by normal non- malicious background traffic. A false positive or false alarm is an alert caused by normal non-malicious background traffic. As shown in Figure 5:3, False Positive Rate of our blockchain cryptographic authentication and verification algorithm detects some normal node as malicious node.

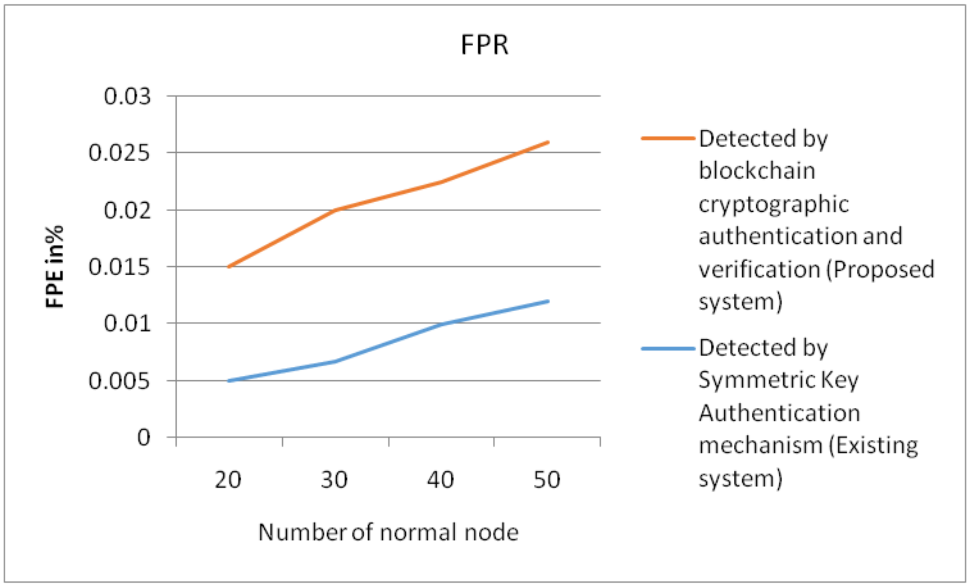


Figure 5-3: False Positive rate of proposed system

5.2 Network Performance of Proposed Method

Network performance is mainly measured in three evaluation metrics which are average end-to-end delay (AE2ED), packet delivery ratio (PDR), and average throughput (TP). In this study we use a Random Waypoint mobility model for nodes movement. The simulation is conducted for tree different scenarios various pause times to show effect of pause time on performance of the network. Pause time is a time for which nodes waits on its first destination before moving to other destination.

The performance for each data flow operation is measured for 50 minutes. To analyze the performance of the network in presence of the malicious node with different pause time. We perform the simulation for separating the malicious node from the network time and to monitor and discover the behavior of the packet forwarding node.

5.2.1 Scenario 1: Average End to End Delay with Pause Time

The comparison of the average end-to-end delay among symmetric key authentication and our proposed system Blockchain authentication and verification with different pause times in all scenarios with malicious node. The simulation result shows the end-to-end delay decreases when pause time increases. Since in symmetric key authentication, the averages end-to-end delay is low when we compare it with our proposed algorithm. This proves that our proposed security method can detect and isolate the malicious node and take time to ignore the malicious node and also it takes some time to make the security execution.

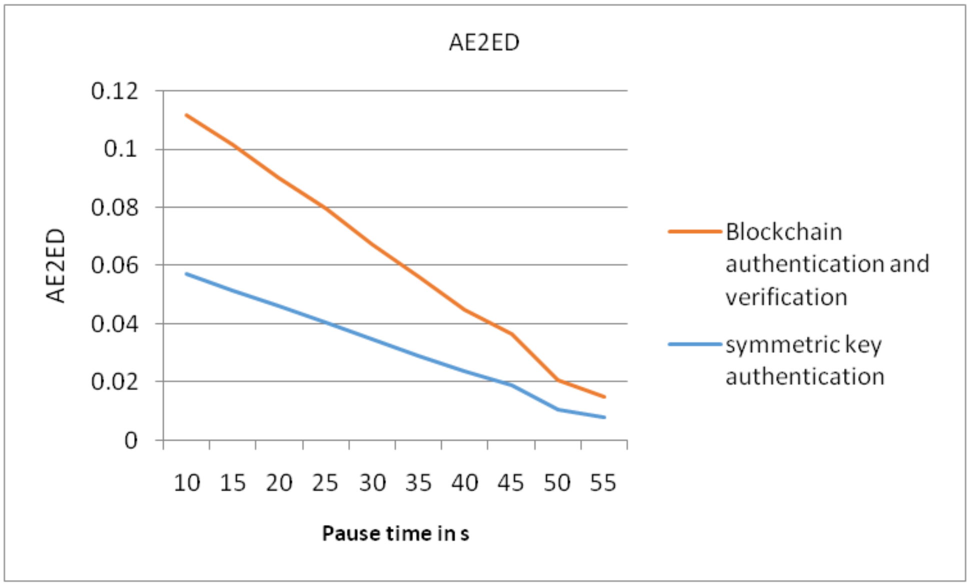


Figure 5-4: Average end to end delay with 20 nodes at various pause time

Figure 5-4 shows the comparison of the average end-to-end delay between our proposed method Blockchain authentication and verification and symmetric key authentication with 20 nodes. The simulation result shows that the average end-to-end delay of the symmetric key authentication mechanism is lower. Based on the result, the difference of averages delay between the symmetric key authentication and our proposed method Blockchain authentication and verification is small. This indicates that there is no significant improvement in terms of delay.

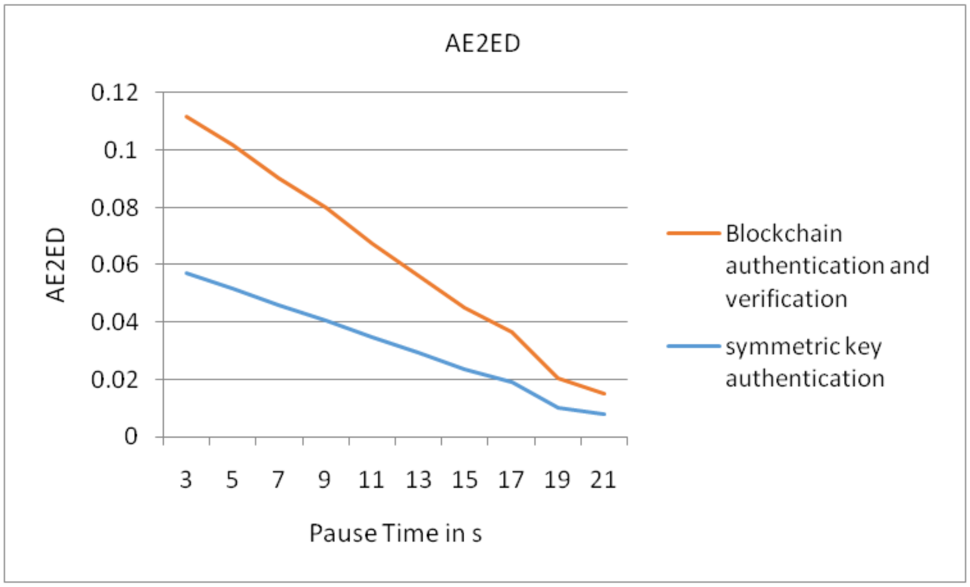


Figure 5-5: Average end to end delay with 30 nodes at various pause time

Figure 5-5 shows the comparison of the average end-to-end delay between symmetric key authentication and our proposed method Blockchain authentication and verification to the variation of pause time to send data when the number of nodes is 30 under a malicious node. The average end-to-end delay of the proposed system and symmetric key authentication difference and symmetric key is better in E2ED.

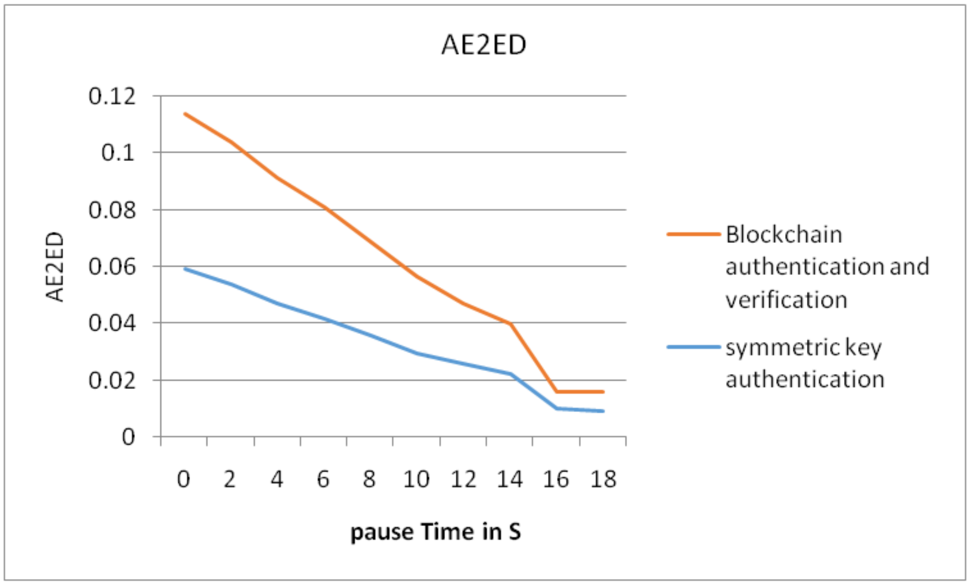


Figure 5-6: Average end to end delay with 50 nodes

Figure 5-6 shows the comparison of the average end-to-end delay between symmetric key authentication and our proposed method Blockchain authentication and verification to the variation of pause time to send data when the number of nodes is 50 under a malicious node. The average end-to-end delay of the proposed system is higher than and symmetric key authentication.

Generally, based on the simulation result in all scenarios, the end of the average to end delay is having some difference and symmetric key authentication is relatively good. But when we see the strength of the security our proposed method Blockchain authentication and verification is higher and strong. This proves that our proposed security mechanism can take a time to detect and isolate the malicious nodes from the network. That also shows the presence of malicious nodes in the network gives a significant effect on the performance of the routing protocol. The result shows that the average end-to-end delay of our proposed method Blockchain authentication and verification is higher than the symmetric key authentication.

Overall in the above three simulation conditions, the average end-to-end delay of symmetric key authentication is better than our proposed method Blockchain authentication and verification with an average advance of 0.031%.

5.2 .2 Scenario 2: Packet Delivery Ratio with Pause Time

Our proposed method Blockchain authentication and verification also evaluated in terms of packet delivery ratio (PDR). PDR is the ratio between the numbers of the delivered data packet to the destination node against the number of the sent packet from the sender node. PDR reflects the network processing quality and data transferring ability. It is the main symbol of reliability, integrity, effectiveness, and quality of the security mechanism. We use the same simulation parameters to evaluate the packet delivery ratio.

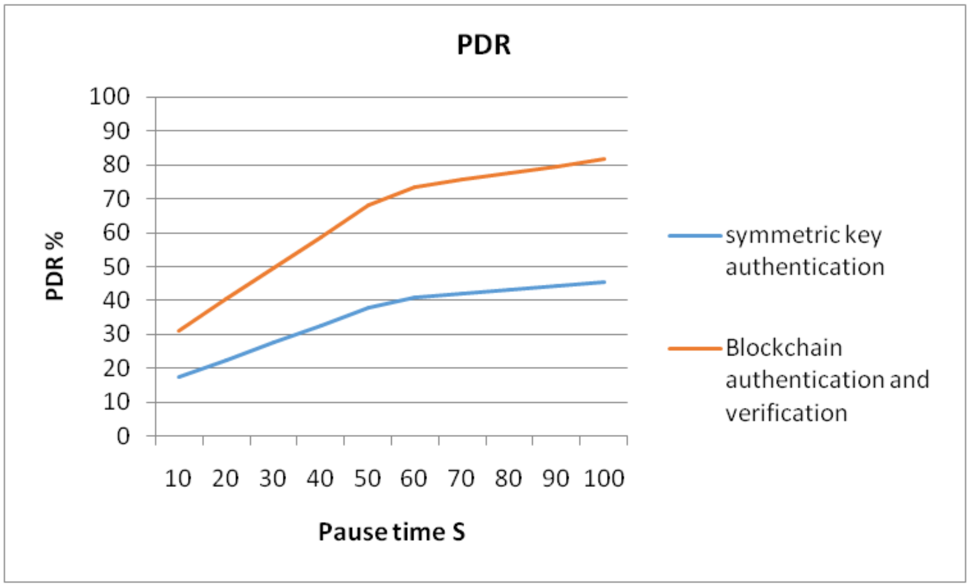


Figure 5-7: Average PDR with 20 node

Figure 5-7 shows when the number of nodes is 20 under malicious node with variable pause time and result show with symmetric key authentication the percentages of the average PDR is low. When we compare the proposed method Blockchain authentication and verification and symmetric key authentication, the percentage of average PDR is high and more than 80%. This proves that our proposed method Blockchain authentication and verification can ignore the malicious node in the communication process of data exchange. The average PDR between symmetric key authentication and our proposed method Blockchain authentication and verification that the average PDR of Blockchain authentication and verification is better than the symmetric key authentication method.

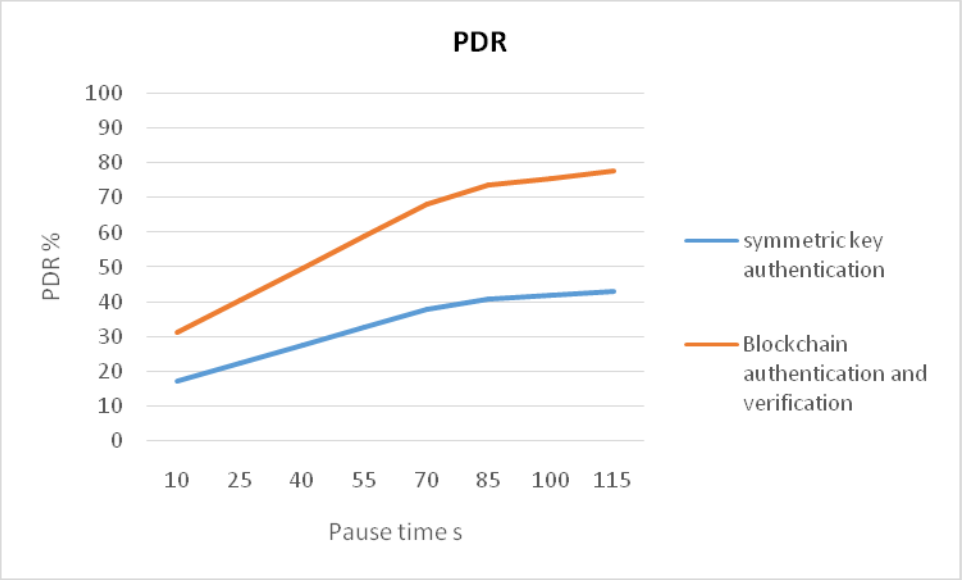


Figure 5-8: Average PDR with 30 nodes

Figure 5-8 shows PDR between symmetric key authentication and our proposed method Blockchain authentication and verification when the number of nodes is 30 under a malicious node with variable pause time. The simulation results show at a speed of 10m/s with 15 seconds time interval pause time, the average PDR of our proposed method Blockchain authentication and verification is better than symmetric key authentication. Based on the graph, the difference value of average PDR between the symmetric key authentication and our proposed method Blockchain authentication and verification becomes big when the speed is increased.

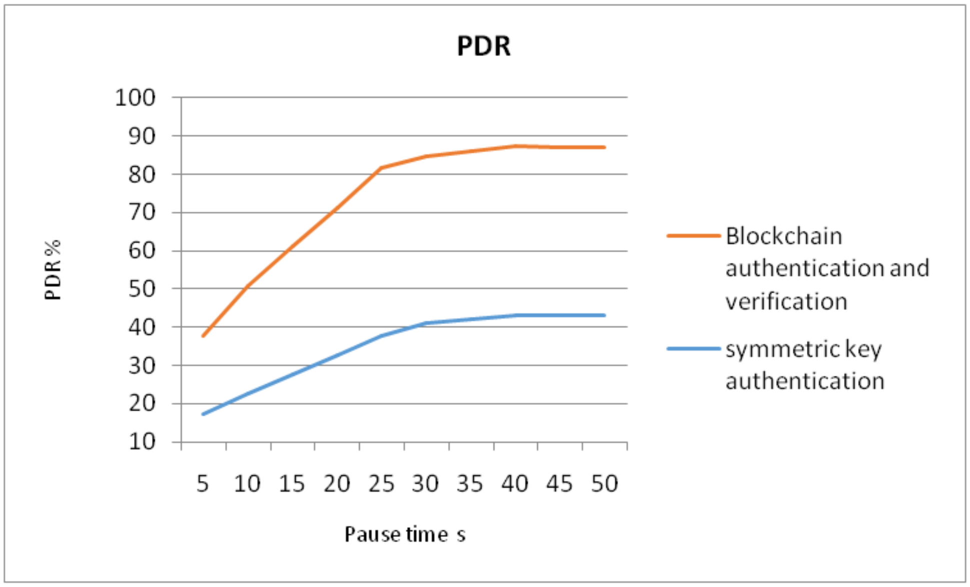


Figure 5-9: The average PDR 50 nodes

Figure 5-9 shows the effect of speed on the average PDR values in symmetric key authentication and our proposed method using 50 nodes under a malicious node with variable pause time. Simulation results show that the average PDR of our proposed method Blockchain authentication and verification is higher than symmetric key authentication, especially in high-speed mobility.

Based on the above graphical record, we can see that the average PDR of symmetric key authentication is small. This indicates that the malicious nodes significantly reduce network performance. In our proposed method Blockchain authentication and verification, the average PDR is high. This also establishes that our security mechanism can identify and ignore the malicious node from the network. When we Comparing the symmetric key authentication and our proposed method Blockchain authentication and verification, the average packet delivery ratio decreases when the speed of node mobility is increased. Since the speed of node mobility increases, the possibility of broken links during the communication process is high due to the rapid change of network topology. A broken link in the network causes many packet losses in the network. It also makes the average PDR values decreases. Simulation results also show that the differences of average PDR values between our proposed method Blockchain authentication and verification and symmetric key authentication increases when the pause time is increased.

5.2.3 Scenario 3: Average Throughput with Mobility Speed

This work last network performance evaluation metric used in our case is average throughput. Throughput is the total number of received bits in the destination in certain time duration. The throughput decreases if many packets are lost in the network.

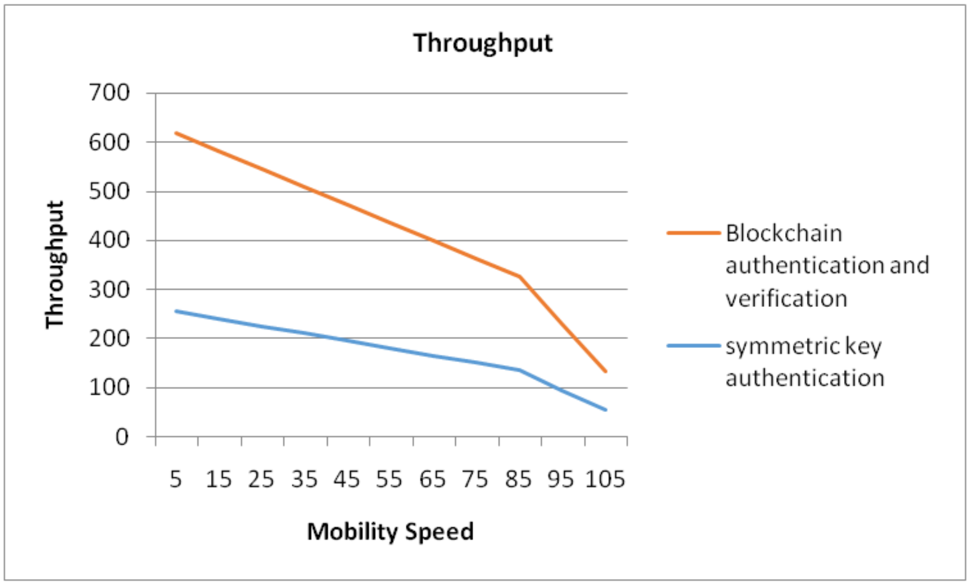


Figure 5-10: Average Throughput With 20 Nodes

Figure 5-10 shows the effect of the malicious node on the average throughput values in symmetric key authentication and our proposed method Blockchain authentication and verification when the number of nodes is 20. Simulation results show that the average throughput of the protocol with Blockchain authentication and verification is better. The trend of average throughput decreases when mobility of node is increased. The averages throughput to our proposed method Blockchain authentication and verification is higher than the symmetric key authentication.

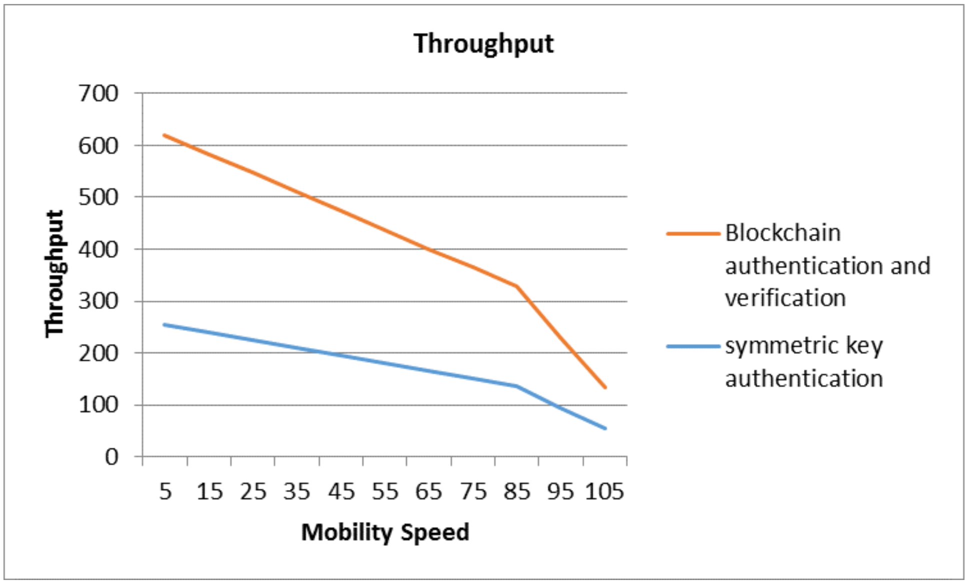


Figure 5-11. Average Throughput 30 Nodes

Figure 5-11 shows the effect of the number of the malicious node on the average throughput values in symmetric key authentication and our proposed method Blockchain authentication and verification when the numbers of nodes are 30. Simulation results show that the performance of symmetric key authentication and our proposed method Blockchain authentication and verification in terms of average throughput is relatively better in packets successfully received in the destination node.

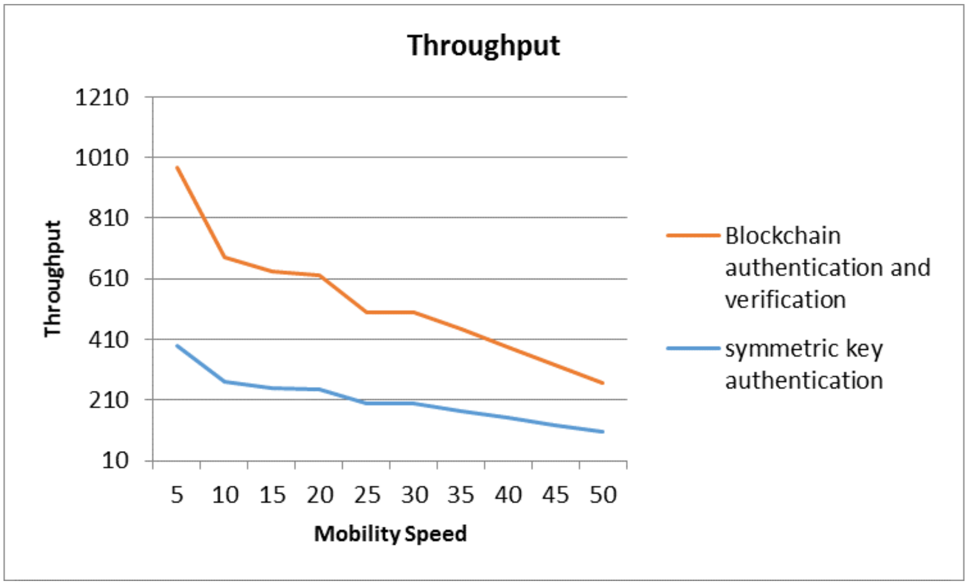


Figure 5-12: The Average Throughput 50 Nodes

Figure 5-12 shows the effect of the mobility of node on the average throughput values in symmetric key authentication and our proposed method Blockchain authentication and verification under a malicious node. Simulation results show that the trend of the average throughput increases when the pause time is decreases. The average throughput of Blockchain authentication and the verification security mechanism is higher than symmetric key authentication. This means that the performance of our proposed security mechanism Blockchain authentication and verification is better in packets successfully received by the destination node.

In all conditions of simulation, the tendency of average throughput increases when the speed of node decreases. Generally, the average throughput of Blockchain authentication and verification is better than symmetric key authentication.

5.3 Summery

In this research work, we improve the performance of symmetric key authentication by modifying it with Blockchain authentication and verification security mechanism in the AODV routing protocol. The implementation of the algorithm is done by adding Blockchain-based authentication and verification for strong security of data exchange and to improve the network performance in MANET.

Our proposed method Blockchain authentication and verification algorithm are evaluated by using NS-2 simulator in terms of security and network performance evaluation parameters. We use Detection Rate, False positive, and false negative for evaluating the security performance of the proposed system to compare it with the base paper of our work. The network performance parameters are end-to-end delay, throughput, and packet delivery ratio. This proposed protocol Blockchain authentication and verification are compared with symmetric key authentication [17] with the appearance of a malicious node to evaluate the security performance and we use different pause time interval to evaluate the network performance of the work.

Simulation results showed that were evaluated for the existing algorithm and the proposed system. The proposed Blockchain authentication and verification were evaluated in terms security performance metrics of detection rate, false negative and false positive. The simulation result of our proposed algorithm shows that the average detection rate of the proposed system is 87.46% and it shows that it is better in detection rate and false negative rate than existing work symmetric key authentication. In the case of network performance, simulation results show that the packet delivery rate and throughput of the Blockchain authentication and verification shows increment in value than Symmetric key authentication increases with different pause time. However, in terms of end-to-end delay, there is no improvement between our proposed method Blockchain authentication and verification and symmetric key authentication. The average packet delivery ratio increases by 7.98%, and the average throughput increases by 4.61%. However, the average end-to-end delay value decreases by 0.031%.

Chapter Six

Conclusions and Future Works

6.1 Conclusions

This research proposed a mechanism to improve the network security and network performance of the MANET network. In our work, the routing approach in mobile ad-hoc networks concerning security is considered and analyzed in mobile ad-hoc networks and proposed the requirements which are essential to be addressed for secure routing. In this paper, we use blockchain authentication and verification algorithm for providing data integrity and other security requirements. Our proposed protocol reaches a better result towards accomplishing the security goals such as message integrity and message authentication, by taking a unified approach of secure Hashing. The other part being the implementation of the Blockchain authentication and verification is that considers the malicious nodes of the network and tries to avoid them as these nodes affect the network performance.

Blockchain authentication and verification work by authenticating every incoming packet and verify the node and its messages by selecting the first key from the key table and generating the Digest1 using MD5. The second key was created by SHA 1 algorithm then the whole message and hope count, the messages seed value digest1 and Digest2 make a block. This block is forwarded to the next node. Then the intermediate nodes first it verifies the message by generating only the Digest1 of the message using the key (The first key = Key number (Hop count mode 15) and if new Digest1 equals to the received Digest1, then the message is treated as from valid node and then again Digest1 is created using another key in the key table and created digest1 is appended into the previous block and the chain of the blocks of the message forwarded to the next node If new Digest1, does not equal the Received Digest1 then message is discarded and the node is rejected from the network then put the node in the rejected list by its ID. At the receiver node, the node generates new Digest 1 and Digest2 if both new digests equal to the received digest, then a message is received as a valid message. If the digest is not equal the message is discarded as an invalid message and the node is rejected from the network. The receiver node uses the Markel tree algorithm to verify all blocks are from genuine nodes. Furthermore, mobility plays an important role while analyzing the network. If the pause time is increased, the mobility decreases that leads to more stable networks. And there are some safeguards against any attack on data confidentiality.

Blockchain authentication and verification are evaluated by using NS-2. The algorithm was evaluated in terms of Security performance parameters detection rate, false negative, false positive. The proposed Blockchain authentication and verification performs better in terms of detection rate, false-negative rate, and false-positive rate than existing work symmetric key authentication. Based on The simulation result of our proposed algorithm shows that its detection rate is 87.46%.

In terms of network performance, we use end-to-end delay, throughput, and packet delivery ratio. This proposed protocol is compared with symmetric key authentication under malicious nodes with different pause time. Simulation results show that the packet delivery ratio of the blockchain authentication increases when pause time increases and throughput increase when mobility speed decreases. However, in terms of end-to-end delay, there is no improvement. The average packet delivery ratio increases by 7.98%, and the average throughput increases by 4.61%. However, the average end-to-end delay value decreases by 0.031%.

Generally, simulation results verified that the implemented security method effectively detects and prevents malicious nodes in the MANET.

6.2 Future Work

In the future, there are some issues to improve in our proposed secure mechanism which are we can’t do in this work

- ✓ Evaluate blockchain authentication and verification security with the different types of attack especially DOS and DDOS.
- ✓ Several other security threats affect other ad-hock networks like VANET FANET and other ad-hoc networks. We recommend that the proposed method to be implemented and tested in the future.

Reference

[1] Liu, H., Zhang, Y. and Yang, T. (2018). Blockchain-Enabled Security in Electric Vehicles Cloud and Edge Computing. *IEEE Network*, 32(3), pp.78-83.

[2] Preeti Sachan and Pabitra Mohan Khilar (2011). Securing AODV Routing Protocol in MANET Based on Cryptographic Authentication Mechanism. *International Journal of Network Security & Its Applications*, 3(5), pp.229-241.

[3] Puthal, D., Malik, N., Mohanty, S., Kougianos, E., and Yang, C. (2018). The Blockchain as a Decentralized Security Framework [Future Directions]. *IEEE Consumer Electronics Magazine*, [online] 7(2), pp.18-21.

[4] R. Kumar, S. Tripathi, and R. Agrawal, "A secure handshaking aodv routing protocol (SHS-AODV)," *2018 4th International Conference on Recent Advances in Information Technology (RAIT)*, Dhanbad, 2018, pp. 1-5.

[5] Aggarwal, R. (2018). A Survey to Improve the Network Security with Less Mobility and Key Management in MANET.

[6] P. Papadimitratos and Z. J. Haas. (2006.) Secure data communication in mobile ad-hoc networks. IEEE Journal on Selected Areas in Communications.

[7] H. Yang, H. Luo, F. Ye, S. Lu, and L. Zhang. Security in mobile ad-hoc networks: Challenges and solutions. Wireless Communications, 2004.

[8] L. Zhou and Z. J. Haas. Securing ad-hoc networks. IEEE Network, 13(6):24–30, 1999. [9]. A. M. Hegland, E. Winjum, S. F. Mjølsnes, C. Rong, Ø. Kure, and P. Spilling. A survey of key management in ad-hoc networks. IEEE Communications Surveys & Tutorials, 2006.

[9] Z. Haas and M. Pearlman. zone routing protocol (ZRP): A framework for routing in hybrid ad-hoc networks. In C. E. Perkins, editor, Ad-hoc Networking, Addison-Wesley, 2001

[10] H. Yih-Chun and A. Perrig. A survey of secure wireless ad-hoc routing. IEEE Security & Privacy Magazine, 2004

[11] K. Sanzgiri, D. LaFlamme, B. Dahill, B. N. Levine, C. Shields, and E. M. Belding-Royer. Authenticated routing for ad-hoc networks. IEEE Journal on Selected Areas in Communications, 2005

[12] S. Yi, P. Naldurg, and R. Kravets. Security-aware ad-hoc routing for wireless networks. In Proceedings of the 2nd ACM International Symposium on Mobile Ad-hoc Networking and Computing (MobiHoc ’01), Long Beach, CA, October 2001.

[13] S. Gupte and M. Singhal. Secure routing in mobile wireless ad-hoc networks. Ad-hoc Networks, July 2003.

[14] Y. Shibata, H. Yuze, T. Hoshikawa, K. Takahata, and N. Sawano. Large Scale Distributed disaster information system based on MANET and overlay network. In Proceedings of the 27th International Conference on Distributed Computing Systems Workshops (ICDCSW'07), Toronto, Canada, June 2007

[15] [6] Ejaz, W., & Anpalagan, A. (2019). Blockchain Technology for Security and Privacy in Internet of Things. In the *Internet of Things for Smart Cities* (pp. 47-55). Springer, Cham.

[16] Manzoor, A., Liyanage, M., Braeke, A., Kanhere, S. S., & Ylianttila, M. (2019, May). Blockchain-based Proxy Re-Encryption Scheme for Secure IoT Data Sharing. In *2019 IEEE International Conference on Blockchain and Cryptocurrency (ICBC)* (pp. 99-103). IEEE.

[17] Malhotra, Sachin, and Munesh C. Trivedi. "Symmetric Key Based Authentication Mechanism for Secure Communication in MANETs." *Intelligent Communication and Computational Technologies Lecture Notes in Networks and Systems*, 2017, pp. 171–180., doi:10.1007/978-981-10-5523-2_16.

[18] Ramkumar, D., et al. "Continuous Authentication Consoles in Mobile Ad-hoc Network (MANET)." *Cluster Computing*, vol. 22, no. S4, 2017, pp. 7777–7786., doi:10.1007/s10586-017-1386-2.

[19] Malik, Nisha, et al. "Blockchain-based Secure Identity Authentication and Expeditious Revocation Framework for Vehicular Networks." *2018 17th IEEE International Conference On Trust, Security And Privacy In Computing And Communications/ 12th IEEE International Conference On Big Data Science And Engineering (TrustCom/BigDataSE)*, 2018, doi:10.1109/trust com/bigdatase.2018.00099.

[20] J.Viba Mary et al, "A Study on MANET and its Security Concepts." 2019 International Journal of Computer Science and Mobile Computing, Vol.8 Issue.10, October-, pg. 159-163

[21] Joshi, Praveen. "Security Issues in Routing Protocols in MANETs at Network Layer." *Procedia Computer Science*, vol. 3, 2011, pp. 954–960., doi:10.1016/j.procs.2010.12.156.

[22] Bruzgiene, Rasa & Narbutaitė, Lina & Adomkus, Tomas. (2017). MANET Network in Internet of Things System. 10.5772/66408.

[23] Yaga, Dylan & Mell, Peter & Roby, Nik & Scarfone, Karen. (2019). Blockchain Technology Overview.

[24] Rui Zhang, Rui Xue, and Ling Liu. 2019. Security and Privacy on Blockchain. ACM Comput. Surv. 1, 1, Article 1 (January 2019)

[25] Yingli Wang, Jeong Hugh Han, Paul Beynon-Davies, (2018) "Understanding blockchain technology for future supply chains: a systematic literature review and research agenda", Supply Chain Management: An International Journal, https://doi.org/10.1108/ SCM-03-2018-0148

[26] F. Gierschner, Bitcoin and beyond

[27] M. Han, Z. Duan, and Y. Li, Privacy issues for transportation cyber-physical systems, in *Secure and Trustworthy Transportation Cyber-Physical Systems*, Springer, Singapore, 2017, 67–86

[28] J. Li, Z. Cai, J. Wang, M. Han and Y. Li, Truthful incentive mechanisms for geographical position conflicting mobile crowdsensing systems, *IEEE Transactions on Computational Social Systems*, (2018),

[29] Neha Yadav, Urvashi Chug (2019), Secure Routing in MANET International Conference on Machine Learning, Big Data, Cloud and Parallel Computing (Com-IT-Con), 14th -16th Feb 2019

[30] R. Shrestha et al., (2019) A new type of blockchain for secure message exchange in VANET, Digital Communications and Networks, https://doi.org/10.1016/j.dcan.2019.04.003

[31] Subhrajit Majumder, Akshay Mathur, and Ahmad Y. Javaid, A Study on Recent Applications of Blockchain Technology in Vehicular Adhoc Network (VANET), Springer Nature Switzerland AG 2020 https://doi.org/10.1007/978-3-030-31239-8_22

[32] Boudguiga, Aymen, et al. "Towards better availability and. accountability for IOT updates by means of a blockchain." Security and Privacy Workshops (EuroS&PW), 2017 IEEE European Symposium on. IEEE, 2017.

[33] Dorri, Ali, Salil S. Kanhere, and Raja Jurdak. "Towards an optimized blockchain for IoT." Proceedings of the Second International Conference on Internet-of-Things Design and Implementation. ACM, 2017.

[34] Buccafurri, Francesco, et al. "Overcoming Limits of Blockchain for IoT Applications." Proceedings of the 12th International Conference on Availability, Reliability, and Security. ACM, 2017.

[35] D. Ramkumar, C. Annadurai · K. Nirmaladevi (2017) Continuous authentication consoles in a mobile ad-hoc network (MANET), Cluster Computing, https://doi.org/10.1007/s10586-017-1386-2

[36] Ningthoujam Chidananda Singh, D., & Sharma, A. (2020). Understanding the MANET Security Using Various Algorithms and Types. International Journal of Future Generation Communication and Networking, 13(3), 2687-2691.

[37] Fu, Y., Li, G., Mohammed, A., Yan, Z., Cao, J., & Li, H. (2019, August). A study and enhancement to the security of MANET AODV protocol against blackhole attacks. In 2019 IEEE SmartWorld, Ubiquitous Intelligence & Computing, Advanced & Trusted Computing, Scalable Computing & Communications, Cloud & Big Data Computing, Internet of People and Smart City Innovation (SmartWorld/SCALCOM/UIC/ATC/CBDCom/IOP/SCI) (pp. 1431-1436). IEEE.

[38] Rehmani, M. H., & Saleem, Y. (2015). Network simulator NS-2. In Encyclopedia of Information Science and Technology, Third Edition (pp. 6249-6258). IGI Global.

[39] Merlin, R. T., & Ravi, R. (2019). Novel trust-based energy-aware routing mechanism for mitigation of black hole attacks in MANET. Wireless Personal Communications, 104(4), 1599-1636.

[40] Jain, A. K., Bolle, R., & Pankanti, S. (Eds.). (F2006). Biometrics: personal identification in networked society (Vol. 479). Springer Science & Business Media.

[41] Conti, M., Zhou, J., Casalicchio, E., & Spognardi, A. Applied Cryptography and Network Security.

[42] Kahate Stallings, W., 2017. Cryptography and network security. Boston: Pearson.

[43] Bambara, J. J., Allen, P. R., Iyer, K., Madsen, R., Lederer, S., & Wuehler, M. (2018). Blockchain: A practical guide to developing business, law, and technology solutions. McGraw Hill Professional.

Appendix

Appendix A: modification of header file

```
#ifndef blockchain_H_
#define blockchain_H_
#include "blockchain_pkt.h"
#include "blockchain_rtable.h"
#include <agent.h>
#include <packet.h>
#include <trace.h>
#include <timer-handler.h>
#include <random.h>
#include <classifier-port.h>
#include "arp.h"
#include "ll.h"
#include "mac.h"
#include "ip.h"
#include "delay.h"
#define CURRENT_TIME Scheduler::instance().clock()
#define JITTER (Random::uniform()*0.5)
class blockchain; //Forward Declaration
/* TIMERS */
class blockchain_PktTimer : public TimerHandler {
public :
blockchain_PktTimer(blockchain* agent) : TimerHandler() {
agent_ = agent;
}
protected:
```

```
blockchain* agent_;
virtual void expire(Event* e);
};

/* Agent */
class blockchain : public Agent {
/* Friends */
friend class blockchain_PktTimer;

/*Private Members*/
nsaddr_t  ra_addr_;
//blockchain_state state_;
blockchain_rtable  rtable_;
int  accessible_var_;
u_int8_t seq_num_;
protected :
PortClassifier* dmux_; For Passing Packets Up To Agents
Trace* logtarget_; //For Logging
blockchain_PktTimer  pkt_timer_; //Timer for sending packets

inline nsaddr_t& ra_addr() {return ra_addr_; }
//inline blockchain_state& state() {return state_;}
inline int& accessible_var()  {return accessible_var_;}

void forward_data(Packet*);
void recv_blockchain_pkt(Packet*);
void send_blockchain_pkt();
void reset_blockchain_pkt_timer();
```

```
public:
blockchain(nsaddr_t);
int command(int, const char*const*);
void recv(Packet*, Handler*);
};
#endif /* PROTONAME_H_ */
```

Appendix B: sample TCL file

```
# Define options
set val(chan) Channel/WirelessChannel;           # channel type
set val(prop) Propagation/TwoRayGround;          # radio-propagation model
set val(netif) Phy/WirelessPhy;                         # network interface type
set val(mac) Mac/802_11;                         # MAC type
set val(ifq) CMUPriQueue;                        # interface queue type
set val(ll) LL;                                  # link layer type
set val(ant) Antenna/OmniAntenna;         # antenna model
set val(ifqlen) 50;                              # max packet in ifq
set val(nn) 20;                           # number of mobilenodes
set val(rp) Blockchain AODV;              # routing protocol
set val(x) 500;                           # X dimension of topography
set val(y) 500;                                  # Y dimension of topography
set val(stop) 150;                               # time of simulation end
# initialize global variables
set ns [new Simulator]
set tracefd [open one.tr w]
$ns trace-all $tracefd
set namtrace [open one.nam w]
```

```
$ns namtrace-all-wireless $namtrace $val(x) $val(y)


# set up topography object

set topo [new Topography]


$topo load_flatgrid $val(x) $val(y)

create-god $val(nn)

# Create channel

# configure the nodes

$ns node-config -adhocRouting $val(rp)\

-llType $val(ll) \

-macType $val(mac) \

-ifqType $val(ifq) \

-ifqLen $val(ifqlen) \

-antType $val(ant) \

-propType $val(prop) \

-phyType $val(netif) \

-channelType $val(chan) \

-topoInstance $topo \

-agentTrace ON \

-routerTrace ON \

-macTrace ON \

-movementTrace OFF



# creation of nodes

$node_($i) color green
set n0 [$ns node]
set n1 [$ns node]
set n2 [$ns node]
set n3 [$ns node]
set n4 [$ns node]
set n5 [$ns node]
set n6 [$ns node]
set n7 [$ns node]
set n8 [$ns node]
set n9 [$ns node]
set n10 [$ns node]
set n11 [$ns node]
set n12 [$ns node]
set n13 [$ns node]
set n14 [$ns node]
set n15 [$ns node]
set n16 [$ns node]
set n17 [$ns node]
set n18 [$ns node]
set n19 [$ns node]

# Provide initial location of mobilenodes
$n0 set X_ 270.0
$n0 set Y_ 400.0
$n0 set Z_ 0.0

$n1 set X_ 250.0
$n1 set Y_ 400.0
$n1 set Z_ 0.0
```

```
$n2 set X_ 450.0
$n2 set Y_ 100.0
$n2 set Z_ 0.0

$n3 set X_ 400.0
$n3 set Y_ 480.0
$n3 set Z_ 0.0

$n4 set X_ 330.0
$n4 set Y_ 400.0
$n4 set Z_ 0.0

$n5 set X_ 220.0
$n5 set Y_ 150.0
$n5 set Z_ 0.0

$n6 set X_ 400.0
$n6 set Y_ 300.0
$n6 set Z_ 0.0

$n7 set X_ 200.0
$n7 set Y_ 100.0
$n7 set Z_ 0.0

$n8 set X_ 400.0
$n8 set Y_ 410.0
$n8 set Z_ 0.0
```

```
$n9 set X_ 300.0
$n9 set Y_ 200.0
$n9 set Z_ 0.0

$n10 set X_ 310.0
$n10 set Y_ 220.0
$n10 set Z_ 0.0

$n11 set X_ 410.0
$n11 set Y_ 240.0
$n11 set Z_ 0.0

$n12 set X_ 490.0
$n12 set Y_ 270.0
$n12 set Z_ 0.0

$n13 set X_ 330.0
$n13 set Y_ 230.0
$n13 set Z_ 0.0

$n14 set X_ 230.0
$n14 set Y_ 130.0
$n14 set Z_ 0.0

$n15 set X_ 350.0
$n15 set Y_ 150.0
$n15 set Z_ 0.0
```

```
$n16 set X_ 160.0
$n16 set Y_ 160.0
$n16 set Z_ 0.0

$n17 set X_ 280.0
$n17 set Y_ 290.0
$n17 set Z_ 0.0

$n18 set X_ 430.0
$n18 set Y_ 460.0
$n18 set Z_ 0.0

$n19 set X_ 105.0
$n19 set Y_ 290.0
$n19 set Z_ 0.0

# Set a TCP connection between n1 and n3
set tcp [new Agent/TCP/Newreno]
$tcp set class_ 2
set sink [new Agent/TCPSink]
$ns attach-agent $n1 $tcp
$ns attach-agent $n3 $sink
$ns connect $tcp $sink
set ftp [new Application/FTP]
$ftp attach-agent $tcp
$ns at 10.0 “$ftp start”
```

```
# Set a TCP connection between n5 and n10
set tcp [new Agent/TCP/Newreno]
$tcp set class_ 2
set sink [new Agent/TCPSink]
$ns attach-agent $n5 $tcp
$ns attach-agent $n10 $sink
$ns connect $tcp $sink
set ftp [new Application/FTP]
$ftp attach-agent $tcp
$ns at 10.0 “$ftp start”

# Set a TCP connection between n2 and n4
set tcp [new Agent/TCP/Newreno]
$tcp set class_ 2
set sink [new Agent/TCPSink]
$ns attach-agent $n2 $tcp
$ns attach-agent $n4 $sink
$ns connect $tcp $sink

#defining heads
$ns at 0.0 “$n0 label CH”
$ns at 0.0 “$n1 label Source”
#$ns at 0.0 “$n2 label N2″
$ns at 10.0 “$n2 setdest 785.0 228.0 5.0″
$ns at 13.0 “$n4 setdest 700.0 20.0 5.0″
$ns at 15.0 “$n3 setdest 115.0 85.0 5.0″
# Define node initial position in nam
for {set i 0} {$i < $val(nn)} { incr i } {
```

```
# 20 defines the node size for nam

$ns initial_node_pos $n($i) 20

}

# Telling nodes when the simulation ends

for {set i 0} {$i < $val(nn) } { incr i } {

$ns at $val(stop) “$n($i) reset”;

}

# ending nam and the simulation

$ns at $val(stop) “$ns nam-end-wireless $val(stop)”

$ns at $val(stop) “stop”

$ns at 150.01 “puts \”end simulation\” ; $ns halt”

proc stop {} {

global ns tracefd namtrace

$ns flush-trace

close $tracefd

close $namtrace

exec nam simwrls.nam &

}


$ns run
```

Appendix C: sample block forming code in C++

```
class Block {
constructor(index, previousHash, timestamp, data, hash) {
this.index = index;
this.previousHash = previousHash.toString();
this.timestamp = timestamp;
this.data = data;
this.hash = hash.toString();
}
}
```

Appendix D: sample Blockchain code in C++

```
class Blockchain
{
//the first block in the chain
 constructor()
{
this.chain = [this.createGenesisBlock()];
}
createGenesisBlock()
{
 return new Block(0, "23/09/2020", "Genesis block", "0");
}
```

Appendix E: sample block integrity validation code in C++

```
var isValidNewBlock = (newBlock, previousBlock) => {
if (previousBlock.index + 1 !== newBlock.index) {
console.log('invalid index');
return false;
} else if (previousBlock.hash !== newBlock.previousHash) {
console.log('invalid previoushash');
return false;
} else if (calculateHashForBlock(newBlock) !== newBlock.hash) {
console.log('invalid hash: ' + calculateHashForBlock(newBlock) + ' ' + newBlock.hash);
return false;
}
return true;
};
```

Appendix F: sample Awk script for evaluating performance metrics

```
if (time < simulation_start || simulation_start == 0)
simulation_start = time;
sent ++
s_time[packet_id] = time
}if (action == "r" ) {
total_data_hops += hops
e_time[packet_id] = time
recv ++
agtrecv ++
if(e_time[packet_id]> s_time[packet_id])
{
delay = e_time[packet_id] -
s_time[packet_id]
total_delay += delay}
hdrsize=pktsize % 128
total_pktsize += pktsize-hdrsize
if (time > simulation_end)
simulation_end = time;
}}}
/^d/ && /AGT/ && /-It cbr/ {drop++}
($1~/s/ || $1~/f/) && $19~/RTR/ && ($35~/BlockcahinAODV/ ||
$35~/AODV/ ||
$35~/message/ ){rtpkts ++}END{
#otime1 = tmp_e_time - s_time1
##thput1 = (8*total_pktsize /total_delay)/1000
simtime = simulation_end - simulation_start
thput = ((8*total_pktsize) /simtime)/1000
```

```
if (recv!=0){
pdr = (recv/sent)*100
avgdelay = total_delay/recv
nrtovh = rtpkts/recv
avghop = total_data_hops / recv
print (" "nrtovh"\t"" "avgdelay"\t"" "pdr "\t""
thput"\t""
"avghop)
}Else{
print ("error")}}
```