A Trust-based Algorithm for Secure Transmission in MANET

Thesis submitted in partial fulfillment of the requirements for the award of degree of

Master of Engineering

in

Information Security

Submitted By

Ankit Agrawal

(801433004)

Under the supervision of:

Dr. Anil Kumar Verma

Associate Professor

COMPUTER SCIENCE AND ENGINEERING DEPARTMENT

THAPAR UNIVERSITY

PATIALA – 147004, PUNJAB, INDIA

JUNE 2016

Certificate

I hereby certify that the work which is being presented in the thesis entitled, *"A Trust-based Algorithm for Secure Transmission in MANET"*, in partial fulfillment of the requirements for the award of degree of Master of Engineering in *Information Security* submitted in Computer Science and Engineering Department of Thapar University, Patiala, is an authentic record of my own work carried out under the supervision of Dr. Anil Kumar Verma and refers other researcher's work which are duly listed in the reference section.

The matter presented in the thesis has not been submitted for award of any other degree of this or any other University.

Ankit Agrawal
801433004

This is to certify that the above statement made by the candidate is correct and true to the best of my knowledge.

Dr. Anil Kumar Verma
Associate Professor
Computer Science and Engineering Department
Thapar University, Patiala

Countersigned by

(Dr. Maninder Singh)
Head
Computer Science and Engineering Department
Thapar University
Patiala

(Dr. S. S. Bhatia)
Dean (Academic Affairs)
Thapar University
Patiala

Acknowledgement

Words can't express my gratitude towards my guide, **Dr. Anil Kumar Verma**, Associate Professor, Computer Science and Engineering Department, Thapar University, who has seen me through the entire tenure of the work presented in this thesis. I have been fortunate to have an advisor who gave me the freedom to explore the ocean of research on my own and at the same time guided me throughout the thesis. He is an inspiration in the truest sense of the world.

I am also thankful to **Dr. Deepak Garg** and **Dr. Maninder Singh**, Head of Department, CSED and **Ms. Jhilik Bhattacharya**, P.G. Coordinator, for their constant supervision of the thesis work.

I would also like to thanks my classmates who were always there to offer me, the help and facilities required for the successful completion of my thesis.

Most importantly, I would like to thanks **my parents, my friends, my seniors** and above all the **Almighty** for always steering towards the right direction out of the blue, to help me to stay calm in the oddest of the times and keep moving even at times when there was no hope.

Ankit Agrawal
801433004
ME-IS
(2014-2016)

Abstract

A mobile ad hoc network (MANET) is made up of various mobile nodes defined as free to move anywhere in the network and they communicate with each other wirelessly. MANET is an infrastructure-less network and is also self organized network, where nodes are responsible for every operation performed by them and responsible for their maintenance. Due to the dynamic nature of MANET nodes, they are exposed to various security attacks. To increase the security in MANET, it is vital to evaluate the node's trustworthiness in the network. Trust and reputation based approaches provides an effective and efficient way to identify selfish and malicious nodes in the network.

In trust based schemes, each node calculates the trust of other nodes either direct or indirect trust based information. Routing path is measured with the help of the calculated trust values. It is necessary to use the routing mechanism for transferring the packets from source to destination node. Source node needs to find the packet delivery route before sending the data packets. The route must be trusted from source to destination node. So, the most important task in any kind of routing algorithm is to select the secured/trusted path. We propose a trust-based algorithm to select the most trusted path among all possible paths from source to destination node. Trust-based scheme is considered while designing the algorithm. In the proposed algorithm, trust values are assumed among two nodes in the scale of 1 to 10. The proposed algorithm is implemented in JAVA programming language and the results show direct towards the most secured path based on the trust value.

Keywords: MANET, Security, Trust Management, Most Trusted Path

Table of Contents

List of Figures

List of Tables

CHAPTER 1
INTRODUCTION

1.1 Introduction to Wireless Cellular System:

Wireless mobile network have been widely used in different fields since 1980s. Wireless systems are infrastructure oriented and work with the centralized administration, i.e. a base station. Users connect to these systems with the help of base station or the access point. Fixed infrastructure or centralized oriented network bounds the ability of wireless systems [1]. However, it is difficult to use the technology effectively in the absence of a fixed infrastructure. The deployment of network with the fixed infrastructure is not an easy task. To make the deployment easy of wireless network, wireless system must be infrastructure less. Some wireless system such as a MANET, functions in the infrastructure less scenarios and they provide an easy and quick deployment of the network.

MANET is an independent or self directed wireless system, consist of many mobile nodes which connect and communicate via wireless links [1]. MANET is a dynamic and multi-hop technology, which is compiled with the bandwidth restricted wireless links. Mobile nodes mean that they are free to move anytime, anywhere in the network. Mobile nodes are fitted out with the wireless transmitters and receivers with the help of antennas which may be point to point or Omni directional. MANET nodes work like a router which has many interfaces and each interface uses different wireless technology. So, if a MANET node uses technology A and B for its interfaces, such node can communicate with other MANET nodes using technology A or B.

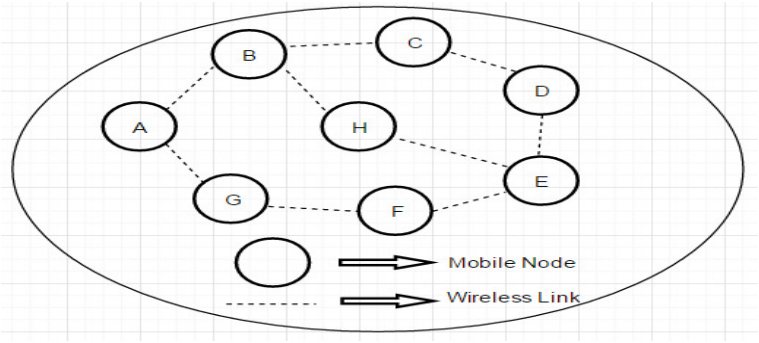

Figure 1.1: MANET

1.2 Motivation:

At the time of designing security protocols, researchers face difficulties because of the limited resources in MANET and characteristics of MANET such as bandwidth, energy, storage, large computation, quick changes in topology due to node mobility or node failure and other wireless characteristics [5].

Trust based approaches and cryptographic approaches are available to provide security in the network. Security can be provided in many ways like authentication of different entities, integrity of messages, confidentiality of messages and resource availability. Security schemes may be applied in different OSI model layers [6]. For example, to provide security at the network layer, purposes of developed schemes are to defend the routing activities against attacks. Such kind of security schemes needs wide computation requirement.

Cryptographic approaches may not be suitable to protect the network from insider attacks because such mechanism assumes the trustworthy and cooperative nature of nodes [3]. The protocols which use cryptographic approaches need to have wide computation and storage. So, there is a new concept of trust and reputation that has been useful on such networks to detect the behavior of nodes and compromised nodes and the same concept is also able to defend the network from the insider attacks. This new trust based concept provides the decision making capabilities to nodes so that they can make decision to forward the packets to valid nodes. Trust based methods are lightweight methods as they need less computation requirement than cryptographic methods [6].

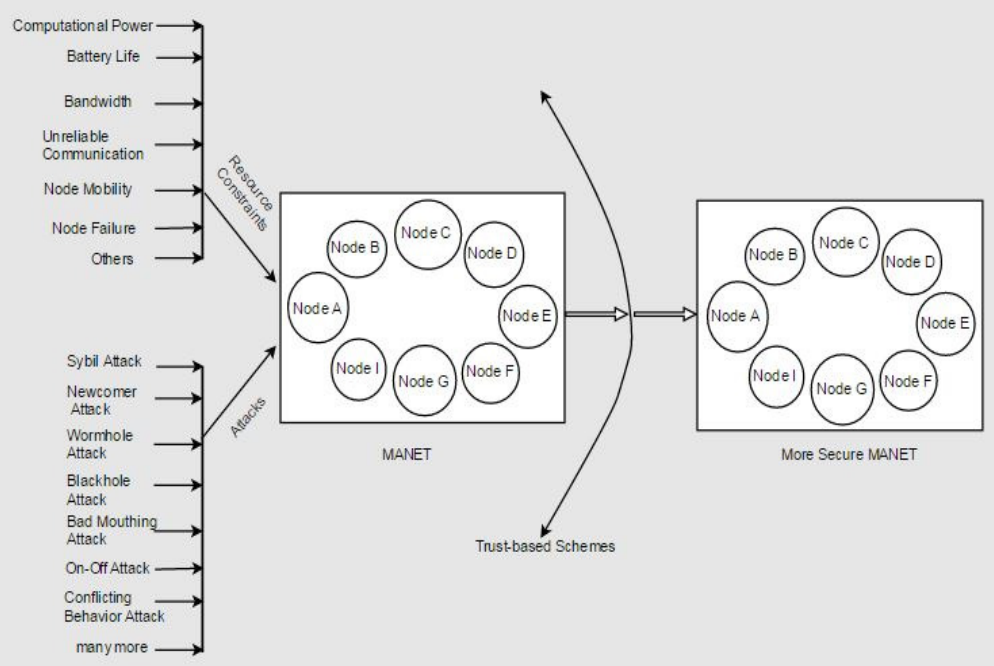

Figure 1.2: Trust Management in MANET

Trust management is a part of the security services; provides and generates security policies and credentials. When nodes want to create the trusted environment in network without any previous interaction, trust management is needed for example bootstrapping. Trust bootstrapping is a way to initialize the trust in the absence of interaction. Trust management has various capabilities to handle the decision making situations like detecting and isolating the misbehaving nodes, intrusion detection, access control and other purposes [5]. To manage the trust is a difficult task in MANET in comparison with the centralized environment. For example, it is hard to gather the trust information due to the changes in the position of nodes. Resource constraint is another factor which creates difficulties in the process of trust evaluation. Uncertainty of trust evidence is also present in MANET due to its characteristics and its dynamic nature.

1.3 Purpose:

The interest of researchers on the trust based security protocols has been increasing extensively. Many trust based schemes have been developed which use different techniques to calculate and manage trust in the network. So, the purpose of this study is to understand different kinds of attacks, the concept of trust and different techniques used by different trust based schemes to compute the trust in the network.

Whenever any node in the network wants to communicate with the other node, it has to first discover the path between them. In the consideration of trust based schemes, it is necessary to find the secure path after calculating the trust value. So, the main objective of this work is to develop a trust-based algorithm to find the most trusted path from source node to destination node in order to secure the transmission in the network.

1.4 Thesis Outline:

The whole thesis is organized into 6 chapters. Chapter 1 is Introduction which describes about the MANET, Motivation, purpose and finally the thesis outline. Chapter 2 is literature Survey which describes about the characteristics and applications of MANET, performance issues in MANET, security issues, different kinds of attacks possible in MANET, concept of trust and various trust based schemes. Chapter 3 is Objective which describes the problem statement and objective. Chapter 4 is Proposed Algorithm which describes about the proposed model and algorithm, an example of the proposed algorithm, pseudo code and the flow chart of the proposed algorithm. Chapter 5 is Implementation and Result which describes about the implementation part and the corresponding results. Finally, in chapter 6, we conclude and discuss the future scope of the work presented.

CHAPTER 2
LITERATURE SURVEY

2.1 Mobile Ad Hoc Network (MANET):

MANET is an infrastructure less system indicates that there is no need of centralized administration to operate and for connecting the wireless users, whereas in such type of systems, each node works as a router or access point for all other nodes in the network. MANET provides flexibility, auto configuration, easier network maintenance, self organizing capabilities, and infrastructure less environment.

MANET performs operations effectively and efficiently in the wireless systems because each node in the network has the capability to perform the routing functionality [2]. Mobile nodes in MANET are self organized and change the topology dynamically [3]. Network topology and number of nodes may change with time due to the mobility characteristics of the MANET. MANET supports host mobility which requires enhancement in the protocol interoperability, but performs network functions such as hop by hop routing by using preexisting routing protocols.

2.2 Characteristics of MANET:

Following are the characteristics of MANET [1][2]-

1) The network does not depend on fixed infrastructure or centralized administration for performing its operations.

2) It provides an easy and quick deployment of the network.

3) It is a multi-hop network.

4) Each node works as an intelligent node such as a router.

5) There is no need of any extra intelligent device for communication.

6) Topology may change dynamically because of the mobile nodes (Nodes may move anytime anywhere in the network).

7) Wireless links are restricted with the bandwidth and each link has its own capacity (variable capacity link) in terms of bandwidth.

8) Nodes are restricted with the energy because they depend on batteries for consuming power.

9) MANET has limited security which means that mobile nodes or MANETs are vulnerable to various kinds of attacks.

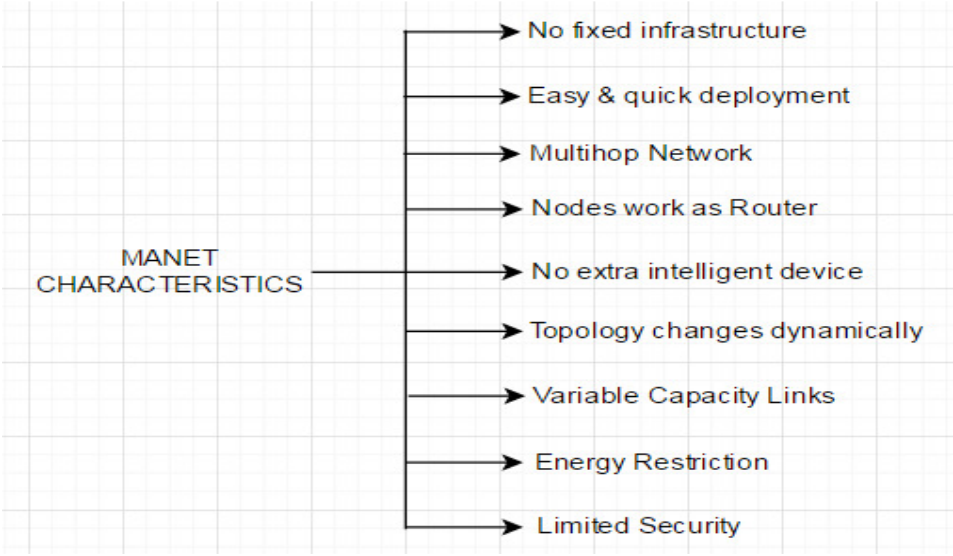

Figure 2.1: MANET Characteristics

2.3 MANET Applications:

There are various kinds of applications developed in MANET such as military surveillance, commercial environment and emergency services etc [3]. There are many applications of MANET technology developed which are used to exchange the mobile data cooperatively such as industrial and commercial applications. MANET technology can be useful to develop applications for fire/rescue/safety operations. MANET technology can be useful in collaborative work such as outdoor meetings, in the home and enterprise networking, such as home networking, conferences, personal area network (PAN), in education such as universities, virtual classrooms, in entertainment such as multi user games [4]. MANET technology can also be useful in the situations where applications require quickly deployment, communication and dynamic networking.

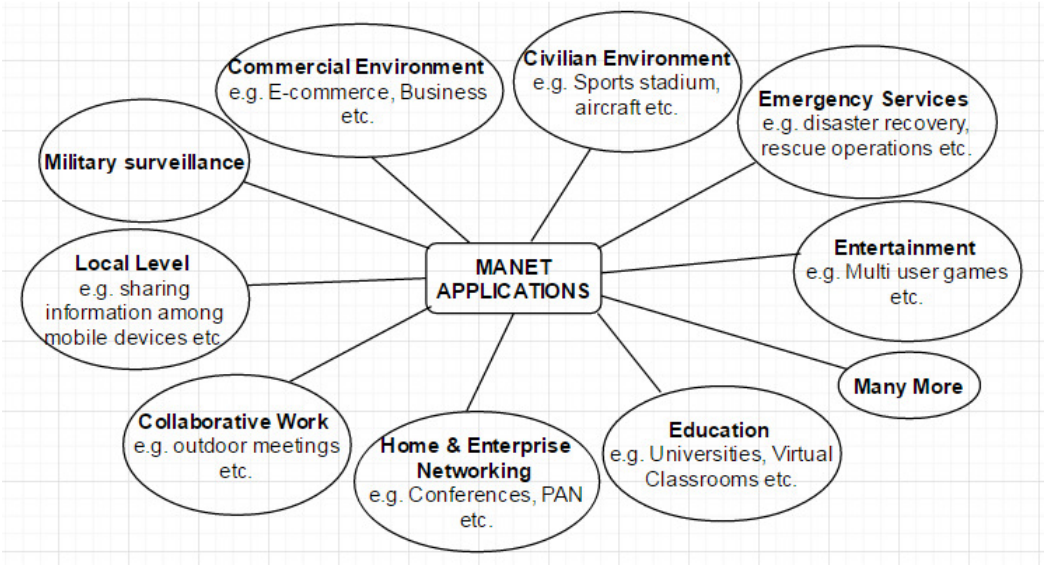

Figure 2.2: MANET Applications

2.4 Performance issues in MANET [2]:

Following are the properties mentioned, which are necessary to be considered to measure the performance of the routing protocol in MANET.

- There is a need to avoid the looping problem in the network. TTL value is useful to solve such problem for getting the best performance.
- Instead of considering the distribution of traffic uniformly, routing algorithm should adjust the traffic on demand basis. It may increase the route discovery delay, but it may help to utilize the network resources efficiently such as energy and bandwidth.
- It is necessary to provide the protection from modifying the routing operation.
- MANET nodes can stop transmitting or receiving for some period of time, which helps to save the node energy. Thus the routing protocol must be capable to fit such sleep period without any consequences.
- Statistical measures such as means, variance and distribution are used to measure the effectiveness of data routing performance.
- One form to measure the end to end delay is to calculate the time required to establish the route.
- One form to measure the routing performance is to calculate the percentage of in order delivery of TCP packets.

- There are some ratios to measure the internal efficiency of a protocol, which are as follows-
  - Average number of data and control bits transmission is used to measure the bit efficiency in the data delivered in the network.
  - Some essential parameters in the context of the network must be considered to measure the performance of protocols like network size, network connectivity and link capacity.

2.5 Security Issues:

The main purpose of providing security in MANET is to protect the network resources from different attacks such as wormhole attacks, Grayhole attack, black hole attack and packet modification attack, etc. Attacker may attack on some node and make them as a selfish node, malicious node and hacker node [6]. Selfish nodes are those nodes which do not forward the packets, but dropped them in order to save their resources i.e. battery. Malicious nodes are those nodes which disrupt the network's operations by performing the attacks such as DoS (Denial of Service) attack, packet dropping, etc. Hacker nodes tap the information communicated over the network by performing wormhole attack etc.

Information sharing and collaboration among nodes are the required operations in MANET. Collaboration is only possible when participating nodes are trustworthy. The nodes in the MANET are deployed in rough or out of control environment so, the harsh environment increases the probability of improper functionality of the network because of the absence of centralized administration [7]. The participating nodes need to be careful while communicating with other nodes as the environmental conditions and node's behavior may change with time. Trust based schemes are suitable to provide the security in such cases as they consider the node's behavior [7]. Network operations depend on the cooperation among nodes. Cooperation between nodes can be destroyed by the selfish nodes and malicious nodes in the network. Without having trust in the network, nodes may fail to perform its task successfully such as sending data in the network, which will result in the degradation of the network performance [8].

2.6 Attacks:

It is a very challenging task to secure a network. It is necessary to understand the functionality of attacks so that a good security mechanism could be established. Any kind of network has some kind of vulnerability. Security depends on the functionality of different applications. Secure network means that each node must be secure enough against the attacks. An attacker may inject malicious things in order to damage the network. Security can be provided to protect the activities in the network such as protect the routing functionality. Two types of nodes may exist in the network named as selfish nodes and compromised or malicious nodes which misbehave.

One thing is that an attacker may convert the compromised node as selfish node. Malicious nodes damage the network intentionally and do not care about its own resources. Cooperation is needed between the nodes in the network for performing successful network functionality such as routing and packet forwarding [1].

2.7 Classification of Attacks:

Attacks in MANET may be classified in different ways, such as internal or insider attack and external or outsider attacks [3]. The outside attacker does not have any authentication in the network, still they may attack using cryptographic approaches, encryption and authentication etc. Whereas insider attacker has all authentication and cryptographic information therefore insider attacks cannot be stopped by using cryptographic approaches. Attacks may be classified as network related attacks and trust related attacks. The main purpose of trust related attacks is to damage the performance of trust based systems so that nodes may not be able to make accurate decisions. For example if misbehaving nodes are present in the network, they may work as intermediate nodes on the routing path and may disrupt the routing functionality in order to damage the network. The main purpose of network related attacks is to damage the overall network performance intentionally by dropping the packets. For example black hole attack drops the packets received from other nodes and such type of attacks may detect by using the trust based schemes. One classification is based on active and passive attacks. Active attacks have intentions to disrupt the network operation by performing the modification, replication of the data communicated over the network. Whereas passive attacks does not have intention to disrupt the network operations.

2.8 Different Attacks in Trusted Environment in MANET:

Following are the attacks which can be performed in a trust oriented environment in MANET.

2.8.1 Wormhole Attack [14]: Wormhole attack is defined as number of malicious nodes cooperating with each other and connects to a slow link known as wormhole link. Malicious node receives packets from source end and sends to destination end and replays these packets. Malicious node may redirect the packet to the slow link created by them so that they could create congestions and delay in the network. Wormhole attack is considered as insider attack.

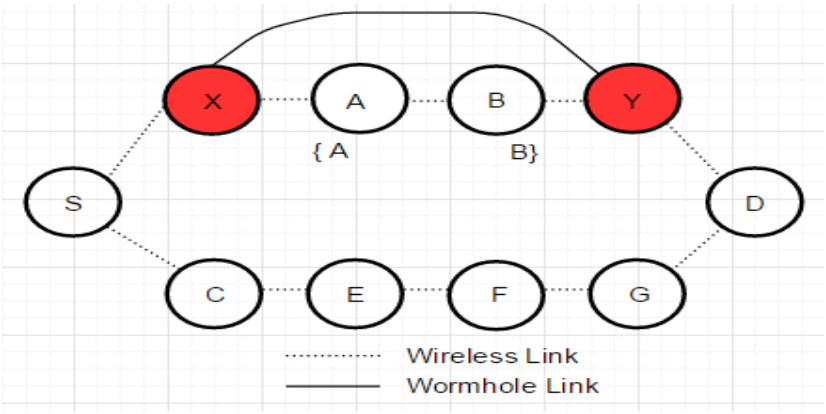

Figure 2.3: Wormhole Attack

In Figure 2.3, node 'S' is represented as source and node 'D' is represented as destination. Node 'X' and 'Y' are malicious node. Node 'A' and 'B' are the good nodes on the route path. Node 'A' and 'B' are both invisible to source and destination node as they come under wormhole. Both malicious node 'X' and 'Y' are visible to both source and destination node. Both malicious node form a tunnel and create a wrong scenario that there is a short path {S-X-Y-Z} between source and destination node. Source will consider the short path instead of considering the long path {S-C-E-F-G-D} and start sending data from the short path. As soon as the malicious node receives the packets, the attacker can compromise or drop them.

2.8.2 Black Hole Attack [10]: Malicious nodes show them as a most perfect node to forward the packets by replying positively to the route request packet, even it does not know where to forward the packets and when they receive packets, instead of

forwarding the packet, they drop them all. Blackhole attack is considered as insider attack.

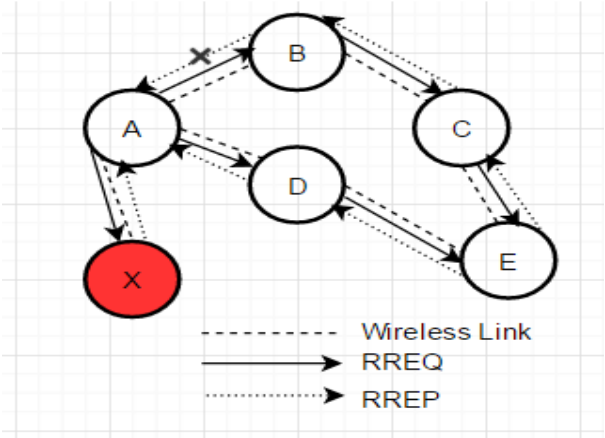

Figure 2.4: Black Hole Attack

In Figure 2.4, node 'A' is represented as source and node 'E' is represented as destination. Node 'X' is a malicious node. Node 'A' wants to communicate with node 'E'. Node 'A' broadcast a route request (RREQ) packet in the network. When node 'X' receives the RREQ packet, it sends back the route reply (RREP) packet to source node immediately. If source node receives a reply packet first from node 'X', it will ignore all the reply packets and start sending data to node 'X'.

2.8.3 Gray Hole Attack [11]: Gray hole attack is a special form of black hole attack. A malicious node attracts other nodes to forward the packets through it. When malicious nodes receive the packets, instead of dropping all packets, they drop selected packets For example; malicious node does not forward the data packets but forwards the routing packets. The Gray hole attack is considered as an insider attack.

2.8.4 Sybil Attack [12]: Attacker represents itself as there is more than one node present in the network while it is actually one node. This is done by using multiple network identities, causing difficult to maintain the network topologies. Sybil attack is considered as insider attack.

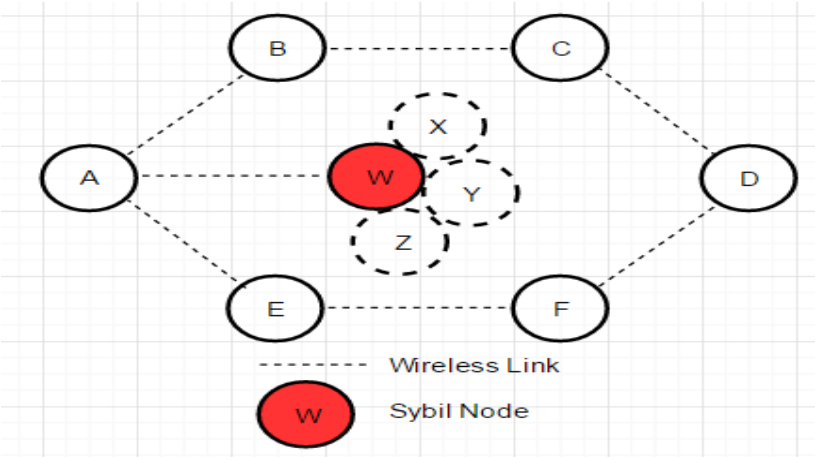

Figure 2.5: Sybil Attack

Node 'A', 'B', 'C', 'D', 'E', 'F' are the good nodes, whereas node 'W' is a malicious node presenting itself with three other identities or nodes e.g. node 'X', 'Y', 'Z'. Whenever a malicious node communicates with any other good node, it creates illusion for the good nodes that good node is communicating with four different nodes. But there is only one node in actual and presenting itself with different identities.

2.8.5 Newcomer Attack [12]: In Sybil attack, attacker node creates several fake IDs so that attacker node can be safe in the network as faked IDs take blame. If Attacker node detects as malicious node, then it leaves the system and enter into the network as a new node with different ID. To do so, attacker node may able to remove its previous bad history in the network.

2.8.6 Routing Loop Attack: In routing loop attacks, the attacker's aim is to stop the packet to reach its destination. Attacker changes the routing information such as number of hops containing in the routing packet so that the packet could not reach its destination but traverse as a cycle in the network. The routing loop attack is considered as external attack.

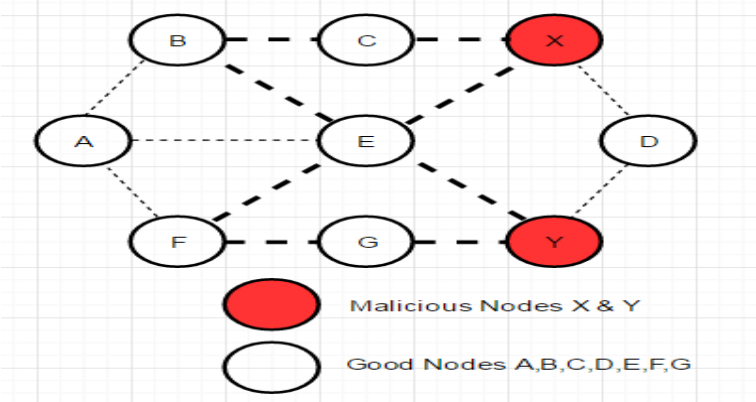

Figure 2.6: Routing Loop Attack

In Figure 2.6, Node 'A' is represented as source and Node 'D' is represented as destination. Node 'X' & 'Y' are malicious nodes. Malicious nodes change the routing information and created loop such as {X-E-B-C-X} and {Y-E-F-G-Y} in order not to reach the packet to the destination.

2.8.7 Conflicting Behavior Attack [17]: Attacker node may behave differently for different set of nodes in the network which conflict with each other. For example, if valid node's opinion is not to send the packet through node A while attacker node says that node A behaves well and the packet must pass through it. In other case, if valid node's opinion is to send the packet through node A while attacker node says that node A does not behaves well and packet should not pass through it. The attacker's aim is to create non trust environment in the network. Conflicting behavior attack is considered as insider attack.

2.8.8 Selective Misbehavior Attack [17]: Attacker node does not perform malicious task to all other nodes but behaves maliciously to some selective nodes. The purpose of such attack is to leave out victim node and make itself as a victim node and receive services from the network.

2.8.9 Bad Mouthing Attack [11]: Attacker's aim is to decrease the trust value of trustworthy nodes in the network. Attacker node provides incorrect information/recommendation by a good node for a trustworthy node so that attacker node can create negative reputation about a good node in order to create a non trustworthy environment in the network. Bad mouthing attack is considered as insider attack.

2.8.10 Selfishness Attack [17]: Selfish node's aim is not to harm the network but is to save its resources such as battery by not allowing forwarding the packets in the network. Selfishness attack is considered as insider attack.

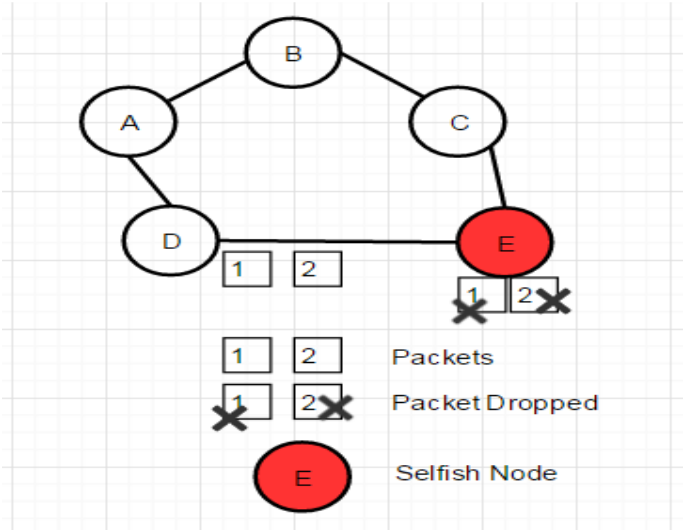

Figure 2.7: Selfishness Attack

2.8.11 On-Off Attack [5]: Attacker node works as trustworthy node (behave well) and untrustworthy node (malicious behavior) on alternate basis so that it can maintain its trust level in the network in orders to not detect as misbehavior node.

2.8.12 Denial of Service (DoS) Attack [15]: Attacker node floods huge number of packets in the network so that it can disrupt the network services and network performance. By flooding the large number of packets, nodes consume excessive resources and not able to communicate further and can block the network services.

2.9 CIA Principle [9]:

CIA stands for Confidentiality, Integrity and Availability, is a security model. These three principles must be present in any system which wants to have security. If one among three principles is breached then the system or concerned party will suffer from the consequences.

2.9.1 Confidentiality: Confidentiality is the first principle which is used to conceal the information from the persons which are not authorized to access it. Confidentiality is achieved by using the cryptography or encryption/decryption methods. It is used to transfer the valuable information from one system to another system.

2.9.2 Integrity: Integrity is the second principle which ensures the accuracy of data. The data are reached to the destination as same as the sender sent. There are various attacks which capture the data and make some changes in the data and then again forward it to the destination.

2.9.3 Availability: Availability is the third principle which ensures that the required data or service or resources are available whenever authorized users want to access them. There are various attacks which flood the bogus data in the network so that the authorized user can not access the resources such as Denial of Service attack.

2.10 Comparison of Attacks in MANET:

Table2.1: Comparison and classification of attacks in MANET

Attacks	Source of Attack	Behavior	Consequences	Security Attribute Affected	Trust Solution Available
Sybil Attack [11][12][13]	Insider	Node uses multiple identities in the network to represent itself as multiple nodes.	Suffer trust evaluation, Affect topology maintenance and fault tolerant schemes, Affect multipath routing	Availability and Integrity	Yes
Newcomer Attack [12]	Insider	Malicious nodes register it as a new node in the network using faked IDs so that it can remove all its bad history.	Suffer trust evaluation	Availability and Integrity	Yes
Worm hole Attack [14][15]	Insider	Make up a false scenario on neighbor relation, capture packet at source-end and tunnel those packets to destination-end and replay those	Threat to the safety of routing protocol, pass on traffic to the time-consuming link to create obstruction and delay	Confidentiality, Integrity and Availability	Yes

		packets.			
Blackhole Attack [10][11][16]	Insider	Attacker claims to have most optimum path to send the packets to the destination node through attacker node so that it can get all the transmitted packets and it can drop all packets	Forged routing information , Degrade the throughput of the network, Wastage of the resource constraints of the network, Threat to the safety of routing protocol	Availability and Integrity	Yes
Grayhole Attack [11][13]	Insider	A type of black hole attack, drops the packets selectively/randomly or modify the packets	Forged routing information , Degrade the throughput of the network, Wastage of the resource constraints of the network	Availability and Integrity	Yes
DoS Attack [5][15]	Insider /Outsider	Malicious node block the normal flow of communication facility by flooding huge number of packets throughout the network	Making off with the hardware, Attempts to penetrate the victim's machine, Block the services for other users	Availability	Yes
Black mailing Attack/ Badmouth[3][11]	Insider	Malicious node blackmails to good node by sending wrong information that other node is misbehaving maliciously which is actually not.	Attempt to change the routing topology, To disrupt the network functionality, To decrease the trustworthy node's trust rating	Availability and Integrity	Yes
Selfishness [17]	Insider	Selfish node does not cooperate to forward the packets for preserving its resource	Misroute packets	Availability	Yes

		constraints so, it drop the packets			
On-off Attack [5]	Insider	Attack node alternatively behaves like trustworthy node and untrustworthy nodes to stay undetected while causing damages	To disrupt the routing protocol	Availability and Integrity	Yes
Selective misbehaving Attack [17][18]	Insider	Attacker misbehaves on selectively victim nodes and behave normally with other nodes who are performing crucial role in the network	To disturb the trust environment, To disrupt the routing path	Availability and Integrity	Yes
Conflicting behavior Attack [17]	Insider	Malicious node behaves differently for different set of nodes to make the recommendation from different set of nodes conflicting that leads to non trusted relationship	To disturb the trust environment, To disrupt the routing path	Availability and Integrity	Yes

Table 2.1, above, describes various attacks on MANET. These attacks have been categorized into two different categories, inside and outside attacks. Each and every kind of attack is also analyzed on the security attribute affected, such as Availability, Confidentiality and Integrity. The table also highlights trust based solutions available for these types of attacks.

2.11 Trust Management:

It is difficult to manage the trust in MANET when cooperation among nodes is to achieve in mission oriented goals such as scalability, availability, reliability etc. To manage trust in mission oriented applications, it is important to consider the resource constraints and other properties of MANET like node mobility and node failure [5]. So there are some trust based schemes which have used the concept of social trust and quality of services trust to obtain the trust metric.

Trust and reputation management are both related terms; there exists a little bit difference between reputation and trust. Trust is defined as the belief of a node about its peer nodes, whereas reputation is defined as the perception that the peer node figures about a node. In many papers, authors have been using cryptographic systems in a MANET to provide the security in terms of confidentiality, integrity and availability. Such methods take much time in computation. Such methods are very useful in providing confidentiality, authentication and integrity, but when the concern is availability, overall throughput and robustness of the network, cooperation and trust between nodes must be exist [6]. Many trusts based protocols have been developed with different purpose, such as to secure routing, access control, intrusion detection, key management, and authentication and to improve resource utilization, to improve network throughput [5][19].

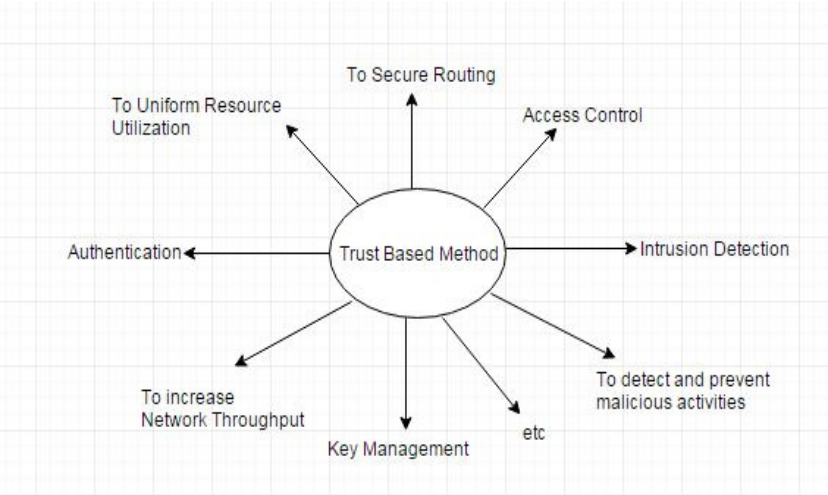

Figure 2.8: Design purposes of Trust-based Methods

2.11.1 Trust Definition:

Trust in a wireless network can be defined as the quality of being reliable of other nodes, which are performing actions [3]. Trust is defined as the belief of a node about its peer node based on its behavior [5]. Trust is defined as a subjective judgment by other peer nodes based on the information passed and action taken through that node according to the context [7]. One node can assign trust to another by analyzing all previous records in terms of behavior in all situations [20]. Trust can be defined as an estimation of evaluating node about the node which is targeted by analyzing the past behavior of the target node known as direct information and recommendation from the intermediate nodes (recommending nodes) among the evaluated and target node known as indirect information [21]. Trust level represents the honesty of a node which helps to decide whether the node is trustworthy or untrustworthy [22]. Trust level is defined as ranges from 0 to 1, 0 indicate complete distrust and 1 indicate complete trust.

2.11.2 Trust Concept:

Trust is an important concept to handle with the uncertainty in MANET. Trust computation and managing the trust is a challenging task due to the resource constraints and node mobility and characteristics of MANET [7]. It is difficult to evaluate trustworthiness from the collecting trust information because of the dynamic environment of the network [23]. The nodes which are untrustworthy can affect the data quality and data reliability. Node's trust value is computed by using some metric or recommendation. The computed trust values propagate in the network to establish the trust among nodes. Trust aggregation is used to get the combined trust value coming from multiple paths. This combined trust value will be used to provide the security in the network.

Trust brings the confidence and belief, and represents relationship among nodes. Experience, recommendation and knowledge are three components of trust computation. The first trust component "experience" for each node is calculated from the past node's behavior and keeps it in trust table. Trust table is considered as recommendation and propagated to all other nodes. The trust which is evaluated previously is included in the current component "knowledge" of total trust. If trust

management is combined with the cryptography, hard security solutions can be provided to the network.

Trust value is measured by analyzing the behavior of nodes in most trust based schemes. Trust measurement will be different based on different parameter for different applications. The performance of trust management system is evaluated by using the system performance metric which includes throughput, overhead, packet dropping rate, delay etc [6].

2.11.3 Trust Properties:

Following are the trust properties [5][19] in MANET

1) Trust is not static but dynamic. Trust based information may be changed as behavior of nodes is not same all the time and may be changed time to time.

2) Trust is subjective. The trustor node may obtain different trust level for the same trustee node, because the trustor node may have different experiences with the node due to the topology changes dynamically.

3) It is not necessary that the trust is transitive. For example, node X trusts node Y, node Y trusts node Z, it is not necessary that node X will trust node Z. If it is necessary to have the transitivity property among two nodes (A-Z) in MANET, third party (Y) gives recommendation about the trustee node (Z) to the trustor node (A).

4) Trust is asymmetric. Nodes which are highly capable in terms of energy and other factors do not trust on the nodes which are less capable. While lower capability nodes may trust on the higher capability nodes.

5) Trust is context dependent. Node X may trust node Y as an unselfishness node but not as a packet modifier node.

Many trust based schemes do not reflect on trust properties. Some trust based schemes use discrete variables to represent the trust while some trust based schemes assume that trust is transitive and symmetric.

2.11.4 Trust Process:

Following are the critical factors which are used to establish trust such as bootstrapping, trust evidence, trust evaluation and decision making [3]. Bootstrapping is the first step to initialize the trust value in trust and reputation system. Following are the three most ways to initialize the trust value

1) When nodes are initialized with high trust value, nodes are treated as trustworthy node.

2) When nodes are initialized with neutral trust value, nodes are neither treated as trustworthy nor non-trustworthy.

3) When nodes are initialized with low trust value, nodes are treated as non trustworthy.

Different trust based models initialize trust value with one of above mentioned values. The trust value may or may not be consistent; increment or decrement in the trust value depends on the node's behavior [20]. The misbehaving nodes with high or low trust value may take time to decrease or increase the trust value respectively. Therefore most trust based models initialize the nodes with neutral trust value. Nodes observe the behavior of other nodes based on their experience and based on the observation they decides the trustworthy of other nodes. In trust based methods, trust evidence is classified into two ways: direct trust and indirect trust. Trust evaluation is performed based on either both or individual trust evidence. When a node interacts with its neighbors directly and calculates the trust of its neighbor by its own experience is considered as direct trust. When a node calculates trust of the node which is not its direct neighbor by having the recommendation from other nodes is considered as indirect trust. Indirect trust is more vulnerable than direct trust as trust rating is recommended by other nodes. Many researchers have been using different kind of trust computation approaches such as beta distribution in probabilistic based approach, Game theory based, weighting based, neural network based, Bayesian based, fuzzy logic based and entropy based approaches. Decision making component of trust and reputation system helps to make selection of trustworthy nodes. Decision making component has been classified into three types: threshold based, weight

based, and ranking based. Some trust based schemes have been using hybrid nature (two of them) of these decision making components.

2.12 Classification of Trust Management:

Trust management is being widely used in detecting and isolating attacker node or selfish node in order to secure routing process, access control, key management and decision making capabilities etc. Trust management has been classified into various aspects [5].

1) Reputation based Vs trust establishment framework: Direct and indirect observations are being used to evaluate a node in reputation based framework. In trust establishment framework, neighboring nodes are evaluated by direct observations and the nodes without any prior interaction are evaluated by the opinions from the intermediate nodes.

2) Policy based Vs reputation based trust management: Security strategies such as logical rules which can verify the users are being used to access the resources in policy based trust management. Computational mechanisms are being used to evaluate the trust in reputation based trust management.

3) Evidence based Vs monitoring based trust management [24]: In evidence based trust management, trust relationship among the nodes can be proved by using the evidences such as identity, address and public key etc. In monitoring based trust management, trust evaluation for any node in the network is done by direct observation (behavior of neighboring nodes) and indirect information (recommendation from other entities).

4) Certificate based framework and behavior based framework: In certificate based framework, trust relationship between nodes is recognized by using the certificates which are maintained by the cooperation among nodes. In behavior based framework, each node evaluates the trust by monitoring the behavior of its neighboring node. It is assumed in behavior based framework that node identification is done by preloaded authentication mechanisms.

5) Hierarchical framework and distributed framework: Nodes create a hierarchy based on their trust level by providing evidence by trusted third party. In distributed

framework, each node has its own responsibility to obtain, distribute and maintain the trust evidence.

2.13 Trust Based Schemes:

Cho, J.H. et. al [19] proposed a trust based protocol for mission-aimed group communication system (GCS) in the MANET and analyze it using hierarchical modeling techniques which are based on stochastic Petri nets. The main purpose of this scheme is to discover the optimal trust chain length by using trust metric which reflects the trust based characteristics and generates accurate trust level for collaboration among nodes in the network with the consideration of trust availability and path reliability.

Author considers the malicious node and selfish nodes when participating nodes do not have prior interaction among them. Author assumes that each node is associated with the IDS to identify the insider attacks. The trust value of each node is calculated by direct observations and indirect information (recommendation by another node). Author uses the symmetric key or group key or secret key for communication among group members. Each node broadcast a message containing secret key and the direct information of its one hop neighbors known as status exchange message so that each node can calculate the trust of other nodes with the help of such information.

Pirzada, A.A. et. al [20] proposed a trust model which is based on individual experiences of nodes (direct trust) and not relying on the recommendation given by a third party. Author considers a variable known as weight, which increase or decrease its value with the experiences of nodes over time.

Each node works as a trust agent in this model. Each agent operates independently in the network and collects data from all events, give weight to each event and finally computes the trust. Each agent performs the three functions: 1) Trust derivation- trust is computed by gathering information from one node about the other node. Possible events or information includes the accuracy of frames received, data/control packets received and forwarded. 2) Quantification phase, the events monitored in the derivation phase is quantized and weights are assigned to the events. 3) Trust computation, the trust value and weights are combined for all the events to find the aggregate trust level.

Shabut, A. et. al [21] proposed a trust model. There is always a risk to have dishonest recommendations sent by the misbehaving nodes through various recommendation based attacks like bad-mouthing, etc. Author uses the clustering techniques to filter out the dishonest recommendation based on the three parameters 1) Number of interactions with the evaluated node 2) information similarity with the evaluated nodes 3) closeness between the nodes.

To make sure the consistency of recommendations, Recommendations are collected over a certain time period. Author used Bayesian statistical approach to calculate trust values and follow a beta probability distribution with two parameters α, β where α indicates the collection of positive annotations in terms of forwarding packets and β indicates the collection of negative annotations in terms of dropping packets. The main purpose of this module is to secure the routing protocol where each node calculates the trust value in the routing path between source and destination.

Al-Karaki, J.N. et. al [22] developed a scheme to enhance the level of cooperation among nodes in a selfish environment. This mechanism is used to detect the selfish nodes and eliminate them. Author used two levels: local and global to enforce the cooperation among nodes. Local level indicates a group of neighboring nodes and global level indicates a group of complete nodes. This method used virtual backbone also known as the Virtual Grid Architecture (VGA). VGA consists of cluster heads (CH) which contains a collection of nodes. The area of the network is separated into a number of zones and each CH represents single square zone. In local cooperation, the nodes make a decision to cooperate based on some parameters like location of the nodes, energy constraints on the nodes, mobility pattern of the nodes.

Two parameters take and give are associated with each node. The parameter take indicates that a node has received a certain amount of help from other nodes in the zone. The parameter give indicates that a node has delivered certain amount of help to other nodes in the zone. In global cooperation, when a node enters into a new zone, it sends an identity message to the other nodes so that a node can identify its neighbors. Each node in the network uses direct observation and reputation message for monitoring the behavior of its neighbors.

Manikandan, S.P. et. al [23] proposed a trust based routing scheme which is based on Trust correlation service (TCS) method. The main purpose of this scheme is to

mitigate the black hole attack in MANET. There are two concepts trust and correlation score implemented over the routing protocol i.e. DSR (Dynamic Source Routing). Trust value is computed for a node and correlation score is calculated for various pairs of nodes. The trust value or internal trust (IT) for each node is calculated based on different factors like node's ability to protect against different types of virus attacks, network attacks and illegal resource consumption.

Correlation score is calculated for each pair of intermediate nodes coming in the route path from source to destination based on different type of factors like level of trust, internal trust among pair and number of packets sent and received. This scheme modifies the optional header of the DSR protocol to store the value of TCS. The proposed scheme used Pearson Product Moment Correlation to measure the correlation. The proposed scheme improves the security and overall throughput of the network.

Patel, V.H. et. al [25] proposed a scheme considering malicious activity of the nodes. The proposed trust model consists of two phases: first one is trust formation and second one is to use trust for routing decisions. In the first phase, trust formation means collecting the statistics of the network like number of packets forwarded, number of packets dropped etc. Trust is calculated based on these network statistics. The collection process is continued and uses these statistics only when the node is coming in the requested routing path. Weightage of each parameter is given based on the parameter effect on the performance of the network.

To establish the route, source node broadcast the route request. As soon as destination node receives the request, it sends back reply packet with the trust value 0 of the paths. All the intermediate nodes receive the reply packet, and calculate their respective trust value based on the network statistics and finally add the trust value to the trust of path. As soon as source node receives the reply packet first time, it stores trust value for the path and path itself, and sends the data packets via that path. In case of receiving multiple route reply, it compares the trust value of the stored path with the trust value of the received path and stores the maximum trust value path. The routing path is updated at regular intervals. The main purpose of this trust based model is to utilize the network resources and to prevent the malicious activity performing in the network.

Khan, M.S. et. al [26] proposed a new strategy to compute the adaptive trust threshold (ATT). Proposed ATT considers the mobility as one of the major factors. Author identifies the factors at each node that affects the trust threshold and developed a mathematical model with the consideration of identifying factors for the computation of ATT.

Malicious nodes are also considered here that may inject, alter or drop the data and control packets. It is also assumed that intrusion detection scheme (IDS) is also presented at each node in the network. When IDS detects any kind of misbehave then it activate the ATT strategy module. Each node calculates the threshold and trust of each neighbor. The result of proposed strategy shows the increment in misbehaving node detection rate and improvement in packet delivery ratio.

Kundu, J. et. al [27] proposed a new trust management scheme which is based on Bayesian methods. This scheme is designed for cluster ad hoc network. The main purpose of this scheme is to minimize the requirement of memory space in order to improve the resource utilization. Trust value is calculated among nodes in terms of integer number ranges from o to 50. 6 bit of memory space is required to store the number from 0 to 50. It shows that 81.25% memory is saved. Trust is calculated at intra-cluster level and inter-cluster level based on direct and indirect (recommendation) information. Cluster head (CH) is more capable than other cluster member (CM) nodes in the cluster. In intra-level cluster, it consists of n nodes, including the cluster head. Cluster head broadcast the request message periodically to n-1 cluster member nodes and cluster head receives their trust value and stores these trust values in matrix form. Indirect trust from cluster head to cluster member is calculated using Bayesian methods. Bayesian method depends on the beta probability density function. In inter-level cluster, Cluster heads set up a virtual path to other cluster head through the gateways because there is no direct link among cluster heads. In inter-level cluster, direct and indirect trust is calculated using the gateway nodes.

Saravanan, S. et. al [28] proposed a scheme named as SKITE. SKITE establish sessions based on the recommended values of neighboring nodes. SKITE uses pair wise share secret key to provide trust value and provide secure session. If each node has a secure trust value key (STv) in the network, that node is considered as secure node. If the node does not have key then that node can't participate in the

communication path. If nodes have related or equally recommendations, they may exchange the key for providing security.

In SKITE, Intrusion detection system categorized the node behavior into different states named as trust secured state, vulnerable state etc. based on its trust profile. Audit data of the network in intrusion detection system describe about the node behavior. If node behavior is not the same according to the states, then that node is considered as an intruder or attacker. The requestor node maintains a recommendation metric recommended by its neighbors and updates that metric table before initiating the route. SKITE has three modules named as trust key generator, recommender node and security session manager. If any node wants to make a request for trust value STv then it sends the request to key generator module which authenticate the node. Recommender algorithm provides the trust key to the node and assigns that node to the session manager. Session manager generates the asymmetric key which is exchanged for ATv value obtained from node and assigned to node as STv.

Ant colony optimization (ACO) is a good technique and used to develop routing algorithms. **Sridhar, S. et. al [29]** proposed ANT based trusted algorithm which is implemented in AODV protocol. In this scheme, only those nodes are considered in the routing path whose have a higher trust level than threshold otherwise those nodes are considered as untrustworthy nodes. This scheme is the optimization of routing using the ACO technique in MANET. Route is decided by the AODV protocol. Trust value is calculated for those nodes which are taking participation in the routing and each node compare its trust value with the threshold. Threshold is defined as the average trust value of neighboring nodes. Finally ACO technique is applied to get the optimized routing path in order to get better performance result. The trust value for each node is calculated based on the way they handle the packets, i.e. (the number of packets transmitted, received and dropped). The nodes dropping the packets are removed from the routing path and then the alternate node is decided by the protocol.

Chen, R. et. al [30] proposed a COI dynamic hierarchical trust management (COI-HiTrust) protocol to achieve the mission oriented task. The proposed protocol detects the misbehaving node. Community of Interest (COI) is partitioned into different subtask group and one subtask group leader (SGL) is assigned to each subtask group. Author uses public key infrastructure (PKI) to generate public/private key pair. PKI

protects the network from outside attackers, but not able to provide protection for insider attackers. COI-HiTrust uses intrusion detection system for insider attackers.

Trust computation is based on direct or indirect information. Direct observation is considered when two nodes are under radio range and neighbors with each other. Each COI member computes the trust value of its other equal level members in the same group. Each SGL computes the trust value of other SGLs. A SGL gathers trust value of each COI member in the subtask group and summarizes the trust value for each COI member. Now commander is gathering the trust value from SGLs and summarizes the trust value for each SGL. Author developed a mathematical model to check the performance of the developed protocol. The mathematical model is based on semi-Markov process and uses stochastic petri net techniques.

Jain, Y.K. et. al [31] developed a new trusted routing protocol by using an intrusion detection system (IDS) and trusted framework in order to secure the routing protocol. Author modified the routing messages and routing table of the AODV protocol with new fields which contains the trust based information. The added fields in the routing table are node's opinion, positive evidence and negative evidence. Author uses recommended opinions and cryptographic scheme to discover the trusted routing path. Node 'A' changes its opinion about other node based on the successful or failed communication using trust updating policy. Trust combination algorithm combines recommended opinions got from the neighbors and generate a new opinion for node 'A'. The proposed scheme increases the trustworthiness of the routing procedure and decrease the computation overhead.

Babu, B.S. et. al [32] proposed a new trust scheme which is based on cognitive reasoning. All nodes are associated with the trust by using Behaviors-Observation-Belief (BOB) model. The direct trust is considered and calculated by observing the behavior of neighboring nodes at the time of forwarding the data. At the time of computation of node's trust, trust fading, penalties and rewards are considered. This scheme detects the malicious node by rewarding the positive behavior of nodes and punishing the negative behavior of nodes. The main purpose of this scheme is to secure the routing process and to generate the more reliable routes. Each node contains the belief database. The belief is considered as evidence and used in the trust calculation.

Cognitive agents are used to perform the cognitive acts which are following; sense the changes in the environment and perceiving information, reasoning (conclude) the sensing information, perform actions to make changes in the environment. Dynamic Trust Mechanism – Dynamic Source Routing (DTM-DSR) protocol is an extension of DSR and used in the proposed scheme. In DTM-DSR, route request and reply messages are modified with some extra fields which contains the trust based information. The cognitive agents are used to compute the trustworthiness of its neighbors by using BOB model. Group of cognitive agents establish the trusted route from source to destination.

2.14 Analysis of Trust-based Methods in MANET:

Table 2.2: Analysis of Trust-based Methods in MANET

Trust Based Model	Trust Computation Method	Attack Consider	Routing Protocol	Design Purposes	Methodology	Simulator
Trust based routing in MANET [25]	Weighted of number of data and control packets dropped and forwarded and remaining energy	Packet dropping, flooding, packet delaying	AODV	To secure routing, prevent malicious activities, uniform resource utilization, increase the network lifetime	Direct trust	Qualnet
Adaptive Trust Threshold Strategy [26]	Adaptive Trust Threshold Strategy. Calculate trust threshold based on the network conditions.	Drop, alter or inject data and control packet, Selective forwarding attack	OLSR	To detect and isolate malicious node, increase packet delivery ratio, increase detection rate, reduce false positives	Average trustworthiness of the direct Neighbor	NS-2
Recommendation	Bayesian statistical	Bad Mouthing	DSR	To filter out the dishonest	Direct and	NS-2

Based Trust Model for MANETs [21]	approach by considering the assumptions that trust values follow beta probability distribution	and Ballot stuffing attacks, collusion attack and selfish nodes		recommendation to secure the routing, reduce the false positive and false negatives in order to increase the network throughput	Indirect Trust	
Trust Scheme for Clustered MANET [27]	Bayesian Method	--	--	To improve resource utilization efficiently (i.e. memory space, computation overhead)	Direct and Indirect Trust	--
Trust Based Key Exchange Security Approch [28]	Pairwise key exchange method based on secured trust value (STv)	Packet Dropping attack, Blackhole,Wormhole attack	AODV	To secure a session among MANET nodes	Recommendation metric value STv	NS-2
ANT Based Trustworthy Routing in MANET [29]	Number of packets dropped, received, transmit and take decision based on the threshold	Packet dropping	AODV	To increase PDR (packet delivery ratio) and decrease delay in order to provide trustworthy routing and to optimized route using ACO	Direct Information	NS-2
Hierarchical trust management of COI groups in MANET[30]	COI-HiTrust Protocol uses trust aggregation and propagation for trust evaluation based on	Bad-Mouthing attack, Ballot-stuffing attack, Sybil and Identity	--	To minimize false negative and positives in order to maximize the mission-critical application performance	Direct and Indirect Trust	NS-3

	weighted sum of social and QoS trust	attack				
Trust based routing to mitigate black hole attack in MANET [23]	Based on node's ability to defend against attacks and unauthorized resource utilization, correlation between every pair of nodes	Black hole attack	DSR	To mitigate black hole attack	Direct Information	OPNET
Trust based AODV for MANET [31]	Based on positive events and negative events and opinion (node's belief towards another node's trustworthiness)	Selfish nodes	AODV	To secure routing protocol	Direct Trust	NS-2
A Cognitive Agents based Approach [32]	BOB (Behavior-Observations-Belief)-model based on cognitive theory	Malicious behavior of nodes such as data dumping, data delaying etc.	DTM-DSR	To detect and prevent malicious and dishonest node in order to get secure routing	Direct Trust	--
Analysis of trust manageme	hierarchical modeling techniques	Selfish node, comprom	--	To identify the most favorable length of a trust	Direct and Indirec	--

nt with trust chain optimization in MANET [19]	based on stochastic Petri nets	ised node and newcomer attack		chain among peers based on trust availability and path reliability	t Trust	
Trust Establishment In Pure Ad-hoc Networks [20]	Weighted sum of data and control packets received and forwarded	Black hole attack, Gray hole attack	DSR, AODV	To establish and manage trust in ad hoc network for selecting trustworthy routes	Direct Trust	NS-2

Table 2.2, above, describes the various methodologies (direct trust/ indirect trust) and the computation method used to compute the trust. AODV being one of the popular routing protocol has been exhaustively studied by researchers and the likewise simulation was performed on the NS-2.

CHAPTER 3
PROBLEM STATEMENT AND OBJECTIVE

To detect and isolate the selfish nodes and compromised nodes, trust and reputation based approaches are more appropriate than the traditional approaches. It is a new way to provide the security in the network. Nodes in MANET provide the routing functionality and packet forwarding services to other nodes. So, based on the network statistics such as number of packets forwarded or dropped by the nodes, trust can be established between the nodes. This is done by observing the node's behavior. Trust computation is classified as direct or indirect trust information.

3.1 Problem Statement:

Whenever the source node wants to send the packets to the destination node, then the source node needs to find the packet delivery route before sending the data packets. Many routing protocols have been developed to transfer the message easily and securely in the network. Once the routing path from source to destination node is requested, all the intermediate nodes in the routing path calculate trust values by using direct or indirect trust information. Based on these trust values, routing path is decided.

3.2 Objective:

The main Objective of this work is to find the most trusted path between a source node and a destination node in order to secure transmission in MANET.

.

CHAPTER 4
PROPOSED ALGORITHM

The network is considered as a graph consisting of vertices and edges. Each vertex in the graph is considered as a node in the network. Each edge in the graph is considered as a link between two nodes in the network. It is assumed that trust based scheme is being used to implement the security in the graph. Each edge is associated with a value known as trust value. Trust value is assumed for each edge in the graph in a scale of 1 to 10. Such trust value is considered as weight between two nodes. So, the graph is a weighted graph. Each graph is considered as bidirectional graph.

4.1 Trusted Path Algorithm:

Our algorithm is based on the "Depth first traversal" algorithm. There is one source node (S) and one destination node (D) and the rest nodes are the intermediate nodes. There are three terms used in the algorithm named as a set, READY and VISITED. The set is made up of four entries <Node, "Path", Trust_Value, Hop>, abbreviated <N, P, T, H>: Node is defined as the processing node, Path indicates a path carrying nodes from source to destination, Trust_Value indicates the aggregate trust value of the path., Hop indicates the number of links between the source node and the destination nodes. Weight (W_i) indicates a trust value between the processing node and its neighboring nodes (N_i). Trust_Value and Hop are initialized with 0. READY will be a stack which will store the set. VISITED is an array which will store value 0 and 1 for each node where 1 indicates that the node has been processed in the path and 0 indicates that the node is free to be processed.

4.1.1 Proposed Model:

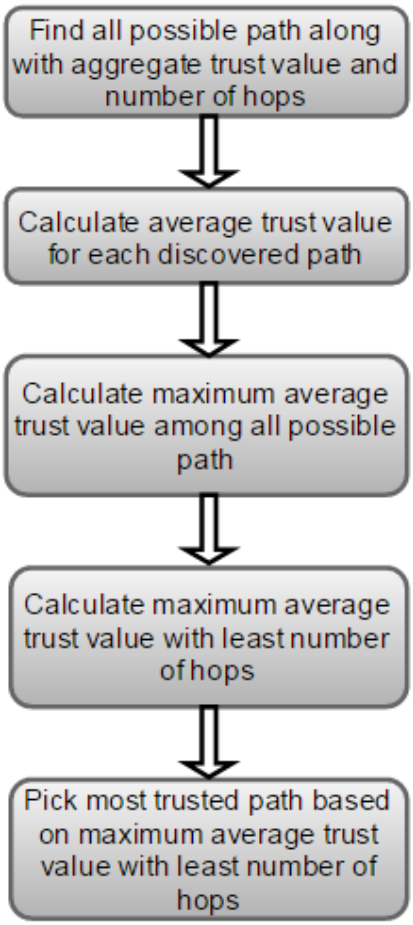

Figure 4.1: Proposed Model

We start from the source node (S) and make a set <S, "S", 0, 0> for the source node and place it into the READY stack. Now we pick one set from the top of the stack as stack is treated as LIFO (Last In First Out) and check if Node in the processing set is same as the destination node, if it is so, then remove the set from the stack and print it, it means that we have got one path from source to destination. Otherwise, process the set and put a value 1 in VISITED array for each node that is present in the path of the processing set and put a value 0 for the rest of the nodes and remove the set from the READY stack. Now we search for all the neighbors of the processing node and creates a set <N_i, 'P-N_i', T+W_i, H+1> for each neighbor if the neighboring node is not yet processed i.e. Visited=0. The created set will be placed in the READY stack.

Now we pick another set from the stack and repeat the process as discussed above until the stack is empty.

This process will give total possible paths from source to destination node, associated with the aggregate trust value (or total trust value) of the path and number of Hops in the path from source to destination. Now the next step is to calculate the average trust value for each path and the formula [33] is used as given below

Final_Trust = Trust_value/Hop;

Final_Trust value is considered as the average trust value. After calculating the Final_Trust for each path, we find the maximum value among the Final_Trust values. Now we do search for the same maximum Final_Trust for more than one path, if it is so, we choose the path which has the least number of hops, and pick the path which has maximum Final_Trust with the least number of hops; is known as the most trusted path in the graph.

4.1.2 Proposed Algorithm:

1) Create a set for source node 'S' i.e. <S, 'S', 0, 0> and place it into the READY stack.

2) Pick one set <N, P, T, H> from the READY stack.

- Check if N is equal to the destination node (D), then remove the set from the READY stack and print the set and go to step 3.
- If not so, process it.
- Remove the set from the READY stack, Visited=1 for those nodes which are in the path 'P' of the processing set and visited=0 for the rest of the nodes.
- Check for all neighbors ' N_i ' of the processing node and check if visited[N_i]==0.
- If so, create set <N_i, 'P-N_i', T+W, H+1> for each neighbor and place them into the READY stack.

3) Repeat the step 2 until READY stack is empty.

4) Calculate Final_Trust for each path as follows

Final_Trust = T/H;

5) Find the maximum value among the Final_Trust values.

6) Check if more than one path has the same Final_Trust value.

7) If so, compare the number of hops for these paths and pick the path which has less number of hops and print the most trusted path of the graph.

A better way to understand the proposed algorithm is to present it with an example.

4.1.3 Example of Trusted Path Algorithm:

Find the most trusted path from the source node '1' to destination node '4'.

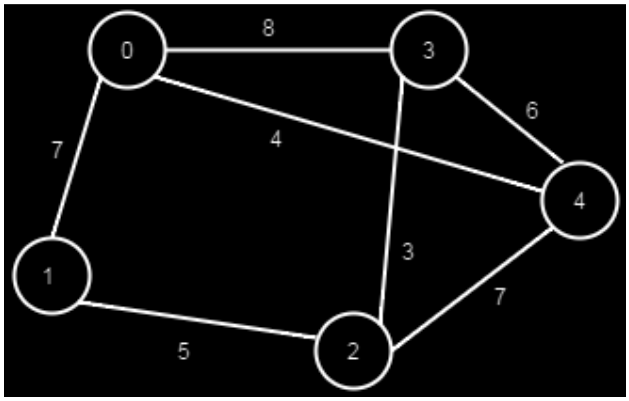

Figure 4.2: Graph1

First step is to create a set for source node and place it into READY stack i.e. READY = <1, "1", 0, 0>

Next step is to pick the set from top of stack which is <1, "1", 0, 0>.

i) Node 1 is not a destination node so, node 1 will be processed, visited [1] = 1. ii) Node 0 & 2 are the neighbors of node 1, visited [0] = 0 and visited [2] = 0. iii) So we will create set for both nodes 0 & 2 and place them into READY stack.	<2, "1-2", 5, 1> <0, "1-0", 7, 1> **READY stack**

Pick another set from top of stack which is <2, "1-2", 5, 1>.

i) Node 2 is not a destination node so, node 2 will be processed, visited [1] = 1 & visited [2] = 1. ii) Node 1, 3 and 4 are the neighbors of node 2, visited [1] = 1, visited [3] = 0 & visited [4] = 0.	<4, "1-2-4", 12, 2> <3, "1-2-3", 8, 2> <0, "1-0", 7, 1>

iii) So we will create set for both nodes 3 and 4 and place them into READY stack.	**READY stack**

Pick another set from top of stack which is <4, "1-2-4", 12, 2>.

i) 4 is a destination node so, remove it from the stack and PRINT the set <4, "1-2-4", 12, 2>.	<3, "1-2-3", 8, 2> <0, "1-0", 7, 1> **READY stack**

Pick another set from top of stack which is <3, "1-2-3", 8, 2>.

i) Node 3 is not a destination node so, node 3 will be processed so, visited [1] = 1, visited [2] = 1 & visited [3] = 1. ii) Node 0, 2 and 4 are the neighbors of node 3, visited [0] = 0, visited [2] = 1 & visited [4] = 0. iii) So we will create set for both nodes 0 and 4 and place them into READY stack.	<4, "1-2-3-4", 14, 3> <0, "1-2-3-0", 16, 3> <0, "1-0", 7, 1> **READY stack**

Pick another set from top of stack which is <4, "1-2-3-4", 14, 3>.

i) Node 4 is a destination node so, remove it from the stack and PRINT the set <4, "1-2-3-4", 14, 3>.	<0, "1-2-3-0", 16, 3> <0, "1-0", 7, 1> **READY stack**

Pick another set from top of stack which is <0, “1-2-3-0”, 16, 3>.

i) Node 0 is not a destination node so, node 0 will be processed so, visited [0] = 1, visited [1] = 1, visited [2] = 1 and visited [3] = 1. ii) Node 1, 3 and 4 are the neighbors of node 0, visited [1] = 1, visited [3] = 1 & visited [4] = 0. iii) So we will create set for node 4 and place it into READY stack.	<4, “1-2-3-0-4”, 20, 4> <0, “1-0”, 7, 1> **READY stack**

Pick another set from top of stack which is <4, “1-2-3-0-4”, 20, 4>.

i) Node 4 is a destination node so, removes it from the stack and PRINTS the set <4, “1-2-3-0-4”, 20, 4>.	<0, “1-0”, 7, 1> **READY stack**

Pick another set from top of stack which is <0, “1-0”, 7, 1>.

i) Node 0 is not a destination node so, node 0 will be processed so, visited [0] = 1, visited [1] = 1. ii) Node 1, 3 and 4 are the neighbors of node 0, visited [1] = 1, visited [3] = 0 & visited [4] = 0. iii) So we will create set for node 3 and node 4 and place them into READY stack.	<4, “1-0-4”, 11, 2> <3, “1-0-3”, 15, 2> **READY stack**

Pick another set from top of stack which is <4, “1-0-4”, 11, 2>.

i) Node 4 is a destination node so, remove it from the stack and PRINT the set <4, “1-0-4”, 11, 2>.	<3, “1-0-3”, 15, 2> **READY stack**

Pick another set from top of stack which is <3, “1-0-3”, 15, 2>.

i) Node 3 is not a destination node so, node 3 will be processed so, visited [0] = 1, visited [1] = 1 & visited [3] = 1. ii) Node 0, 2 and 4 are the neighbors of node 3, visited [0] = 1, visited [2] = 0 & visited [4] = 0. iii) So we will create set for node 2 and node 4 and place them into READY stack.	<4, “1-0-3-4”, 21, 3> <2, “1-0-3-2”, 18, 3> **READY stack**

Pick another set from top of stack which is <4, “1-0-3-4”, 21, 3>.

i) Node 4 is a destination node so, removes it from the stack and PRINTS the set <4, “1-0-3-4”, 21, 3>.	<2, “1-0-3-2”, 18, 3> **READY stack**

Pick another set from top of stack which is <2, “1-0-3-2”, 18, 3>.

i) Node 2 is not a destination node so, node 2 will be processed so, visited [0] = 1, visited [1] = 1, visited [2] = 1 and visited [3] = 1. ii) Node 1, 3 and 4 are the neighbors of node 2, visited [1] = 1, visited [3] = 1 & visited [4] = 0. iii) So we will create set for node 4 and place it into READY stack.	<4, “1-0-3-2-4”, 25, 4> **READY stack**

Pick another set from top of stack which is <4, “1-0-3-2-4”, 25, 4>.

i) Node 4 is a destination node so, remove it from the stack and PRINT the set <4, “1-0-3-2-4”, 25, 4>.	**READY stack**

Stack is now empty and we have got all possible paths from source to destination node.

Next step is to calculate the average trust value (Final Trust) for each path.

Path	**Final Trust**
<4, “1-2-4”, 12, 2>	Final_Trust = 12/2 = 6
<4, “1-2-3-4”, 14, 3>	Final_Trust = 14/3 = 4.67
<4, “1-2-3-0-4”, 20, 4>	Final_Trust = 20/4 = 5
<4, “1-0-4”, 11, 2>	Final_Trust = 11/2 = 5.5
<4, “1-0-3-4”, 21, 3>	Final_Trust = 21/3 = 7
<4, “1-0-3-2-4”, 25, 4>	Final_Trust = 25/4 = 6.25

The maximum average trust value is 7 for the path “1-0-3-4”and there is no other path which has same maximum average trust value. So, the most trusted path is “1-0-3-4” from source node ‘1’ to destination node ‘4’.

4.2 Flow Chart of the Trusted Path Algorithm:

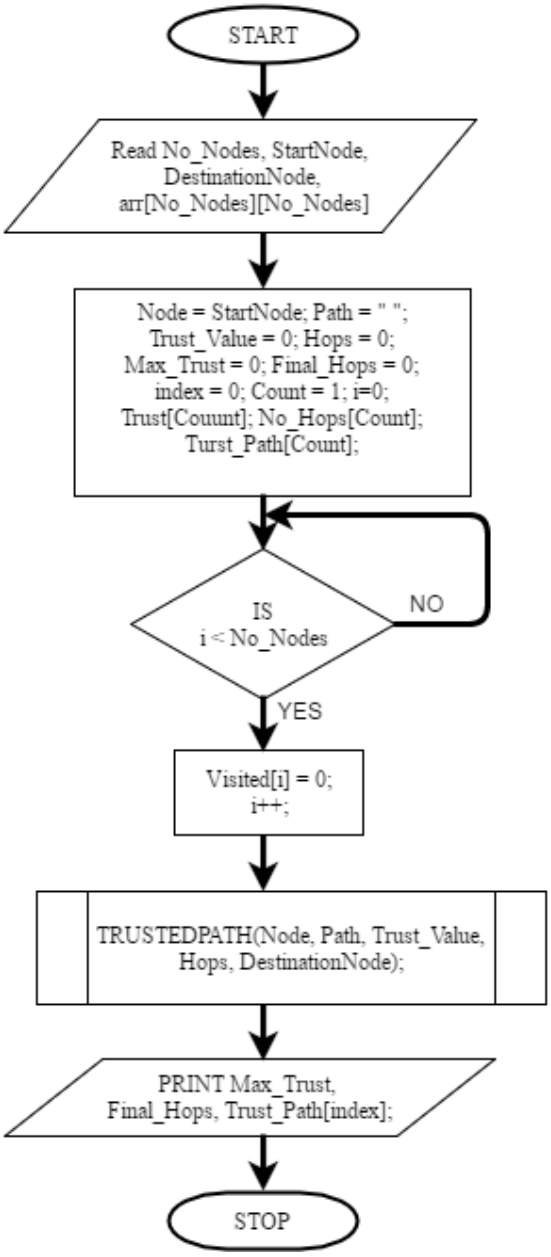

Figure 4.3: Flow chart1 – Main( ) Function

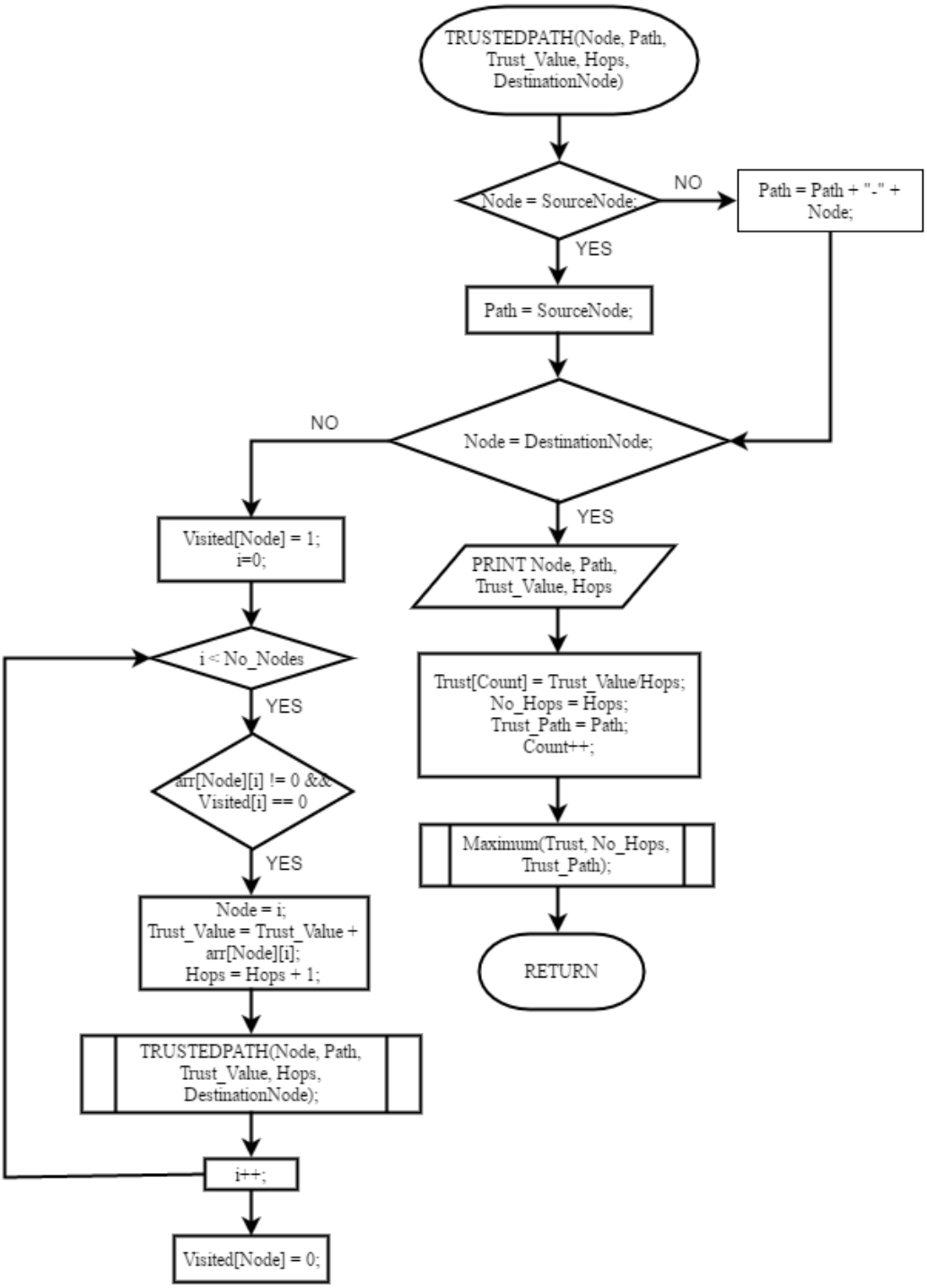

Figure 4.4: Flow chart2 – TRUSTEDPATH( ) Function

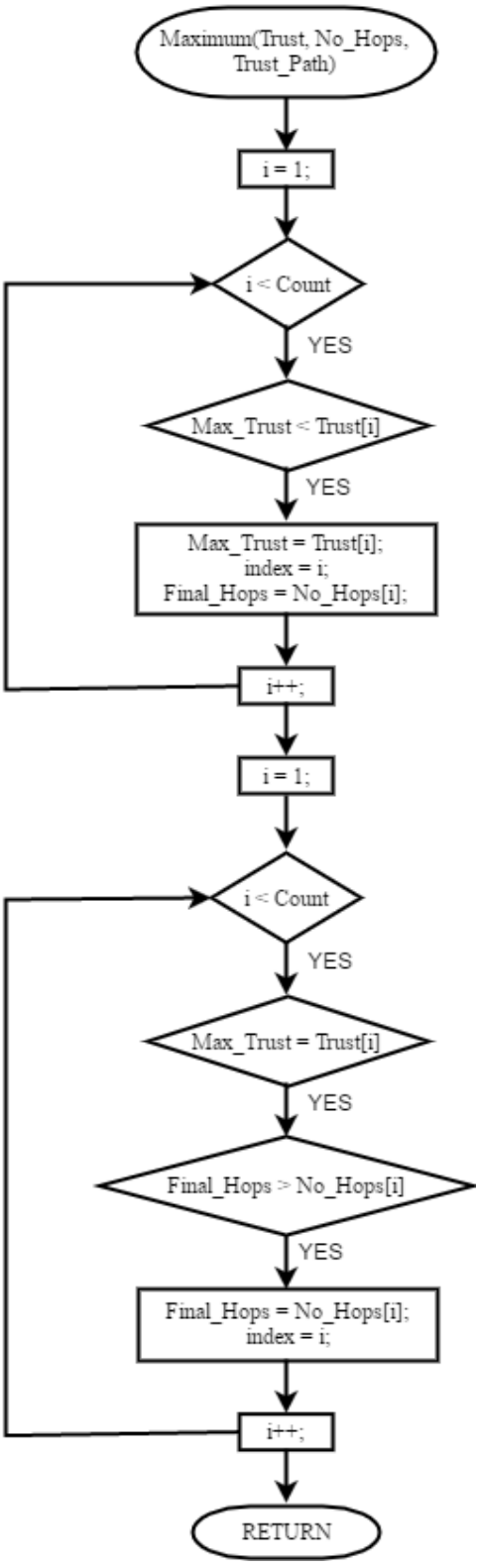

Figure 4.5: Flow Chart3 – Maximum( ) Function

CHAPTER 5
SIMULATION AND RESULTS

5.1 Simulation and Results:

We are using Java programming language to simulate the proposed algorithm in order to get the most trusted path between the source node and destination node. We are simulating the proposed algorithm with different graphs/topologies.

5.1.1 Simulation Steps:

The simulation program performs the following tasks in order to get the most trusted path.

I. Take network input from the user and presenting the network (Graph) as adjacency matrix.

II. Find the entire possible path from the source node to the destination node.

III. Calculate the average trust value for each path.

IV. Find the maximum average trust value among the entire possible path.

V. Find the maximum average trust value with the least number of hops.

VI. Based on the maximum trust value, pick the most trusted path from the source node to the destination node.

5.1.2 Simulation for a Graph:

At first, we are simulating the proposed algorithm with a graph consisting of eight nodes associated with the assumed trust values.

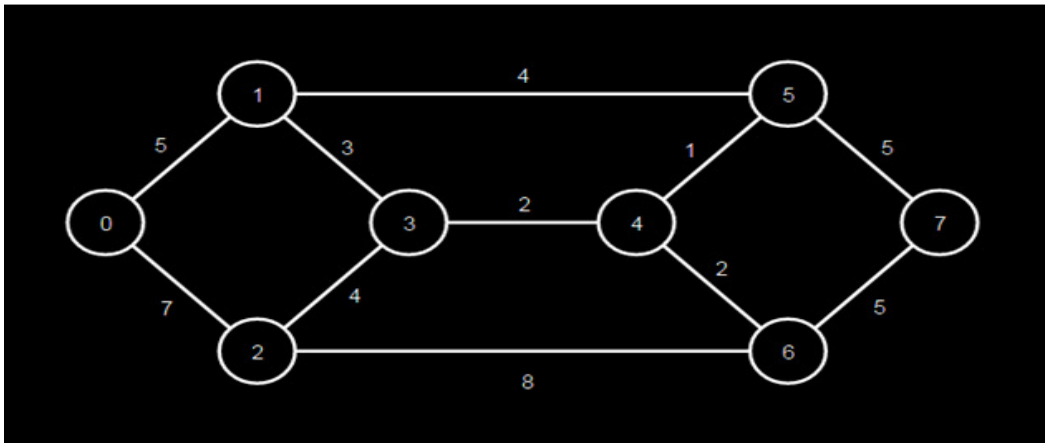

Figure 5.1: Graph2

In the given Graph2, Source node is '0' and destination node is '7'. We are presenting the simulation steps with the screenshots of code and the corresponding result–

I. The simulation code takes input from the user by asking the required parameters like number of nodes required in the graph; number of links required in the graph; and based on the number of links, two node's id and the trust value exist between these two nodes; source node and destination node.

```
System.out.print("Enter the number of nodes in the graph: ");
no_nodes = user_input.nextInt();

System.out.print("Enter the number of links in the graph: ");
no_links = user_input.nextInt();

for(int i=0;i<no_links;i++)
{
    System.out.print("Enter two node number of a link, forwhich you want to enter a value: ");
    node1 = user_input.nextInt();
    node2 = user_input.nextInt();
    System.out.print("Enter a value for the provided link: ");
    arr[node1][node2]=user_input.nextInt();
    arr[node2][node1]=arr[node1][node2];
}

System.out.print("Enter the source node: ");
startNode = user_input.nextInt();

System.out.print("Enter the destination node: ");
endNode = user_input.nextInt();
```

Figure 5.2: Code used to take input from user

The result of the code mentioned in Figure 5.2 is shown in the Figure 5.3

```
Enter the number of nodes in the graph: 8
Enter the number of links in the graph: 11
Enter two node number of a link, forwhich you want to enter a value: 0 1
Enter a value for the provided link: 5
Enter two node number of a link, forwhich you want to enter a value: 0 2
Enter a value for the provided link: 7
Enter two node number of a link, forwhich you want to enter a value: 1 5
Enter a value for the provided link: 4
Enter two node number of a link, forwhich you want to enter a value: 1 3
Enter a value for the provided link: 3
Enter two node number of a link, forwhich you want to enter a value: 2 3
Enter a value for the provided link: 4
Enter two node number of a link, forwhich you want to enter a value: 2 6
Enter a value for the provided link: 8
Enter two node number of a link, forwhich you want to enter a value: 3 4
Enter a value for the provided link: 2
Enter two node number of a link, forwhich you want to enter a value: 4 5
Enter a value for the provided link: 1
Enter two node number of a link, forwhich you want to enter a value: 4 6
Enter a value for the provided link: 2
Enter two node number of a link, forwhich you want to enter a value: 5 7
Enter a value for the provided link: 5
Enter two node number of a link, forwhich you want to enter a value: 6 7
Enter a value for the provided link: 5
Enter the source node: 0
Enter the destination node: 7
```

Figure 5.3: Network input for Graph2

Adjacency matrix is used to represent the graph generated by taking input from the user. Adjacency matrix corresponding to the graph is shown below

```
for(int i=0;i<no_nodes;i++)
    for(int j=0;j<no_nodes;j++)
        arr[i][j]=0;

String str = "|\t";
System.out.println();
System.out.println("Adjacency matrix for a graph is:");
for(int i=0;i<no_nodes;i++)
{
    for(int j=0;j<no_nodes;j++)
    {
        str += arr[i][j] + "\t";
    }
    System.out.println(str + "|");
    str = "|\t";
}
```

Figure 5.4: Code used to represent the Graph2

The result of the code mentioned in Figure 5.4 is shown in the Figure 5.5

```
Adjacency matrix for a graph is:
|       0       5       7       0       0       0       0       0       |
|       5       0       0       3       0       4       0       0       |
|       7       0       0       4       0       0       8       0       |
|       0       3       4       0       2       0       0       0       |
|       0       0       0       2       0       1       2       0       |
|       0       4       0       0       1       0       0       5       |
|       0       0       8       0       2       0       0       5       |
|       0       0       0       0       0       5       5       0       |
```

Figure 5.5: Adjacency matrix for Graph2

II. The next step is to find all possible paths from source node to the destination node.

```
public void TRUSTEDPATH(int nodeNum, String path,int trust,int hops,int dest)
{
    int source=startNode;
    String tpath="";
    if(path.equals(""))
        tpath=Integer.toString(nodeNum);
    else
        tpath=path+"-"+Integer.toString(nodeNum);
    if(nodeNum==dest) {
        System.out.println(source + "\t" + nodeNum + "\t" + tpath + "," + trust+ "," + hops);
        ftrust=((float)trust/(float)hops);
        finalTrust[count]=(float) (Math.round(ftrust*100.0)/100.0);
        finalpath[count]=tpath;
        nohops[count]=hops;
        count++;
        return;
    }
    visited[nodeNum]=1;
    for(int i=0;i<no_nodes;i++)
    {
        if(arr[nodeNum][i]!=0 && visited[i]==0)
        {
            TRUSTEDPATH(i,tpath,trust+arr[nodeNum][i],hops+1,dest);
        }
    }
    visited[nodeNum]=0;
}
```

Figure 5.6: Code used to find all possible paths

The result of the code mentioned in Figure 5.6 is shown in the Figure 5.7

```
Source Dest.    Path,TrutValue,Hops
0       7        0-1-3-2-6-4-5-7,28,7
0       7        0-1-3-2-6-7,25,5
0       7        0-1-3-4-5-7,16,5
0       7        0-1-3-4-6-7,17,5
0       7        0-1-5-4-3-2-6-7,29,7
0       7        0-1-5-4-6-7,17,5
0       7        0-1-5-7,14,3
0       7        0-2-3-1-5-4-6-7,26,7
0       7        0-2-3-1-5-7,23,5
0       7        0-2-3-4-5-7,19,5
0       7        0-2-3-4-6-7,20,5
0       7        0-2-6-4-3-1-5-7,31,7
0       7        0-2-6-4-5-7,23,5
0       7        0-2-6-7,20,3
```

Figure 5.7: All possible paths in Graph2

III. Next step is to calculate the average trust value for each path discovered in the previous step using the following formula

Final_Trust = Trust/Hops;

```
ftrust=((float)trust/(float)hops);
finalTrust[count]=(float)  (Math.round(ftrust*100.0)/100.0);
```

Figure 5.8: Code used to calculate the average trust value for each path

The result of the code mentioned in Figure 5.8 is shown in the Figure 5.9

```
Avg Trust Value         No of hops      Path
    4.0                    7            0-1-3-2-6-4-5-7
    5.0                    5            0-1-3-2-6-7
    3.2                    5            0-1-3-4-5-7
    3.4                    5            0-1-3-4-6-7
    4.14                   7            0-1-5-4-3-2-6-7
    3.4                    5            0-1-5-4-6-7
    4.67                   3            0-1-5-7
    3.71                   7            0-2-3-1-5-4-6-7
    4.6                    5            0-2-3-1-5-7
    3.8                    5            0-2-3-4-5-7
    4.0                    5            0-2-3-4-6-7
    4.43                   7            0-2-6-4-3-1-5-7
    4.6                    5            0-2-6-4-5-7
    6.67                   3            0-2-6-7
```

Figure 5.9: Average trust value for each path in Graph2

IV. Next step is to calculate the maximum average trust value among the discovered path.

```
for(int i=1;i<count;i++)
{
    if(max<finalTrust[i])
    {
        max=finalTrust[i];
        index=i;
        hops=nohops[i];
    }
}
```

Figure 5.10: Code used to calculate the maximum average trust value

The result of the code mentioned in Figure 5.10 is shown in the Figure 5.11

```
Trust Value
    6.67
```

Figure 5.11: Maximum average trust value in Graph2

V. Next step is to calculate the maximum average trust value with the least number of hops if maximum average trust value is same for more than one path.

```
for(int i=1;i<count;i++)
{
    if(max==finalTrust[i])
    {
        if(hops>nohops[i])
        {
            hops=nohops[i];
            index=i;
        }
    }
}
```

Figure 5.12: Code to find maximum trust value with the least number of hops

The result of the code mentioned in Figure 5.12 is shown in the Figure 5.13

```
Trust Value  No of hops
    6.67         3
```

Figure 5.13: Maximum trust value with the least number of hops in Graph2

VI. Next step is to pick the most trusted path from the source node to the destination node based on the maximum average trust value with the least number of hops.

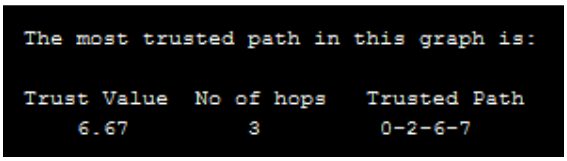

Figure 5.14: Most trusted path in Graph2

5.1.3 Simulation for Mesh Topology:

We are simulating the proposed algorithm with mesh topology consisting of five nodes associated with the assumed trust values.

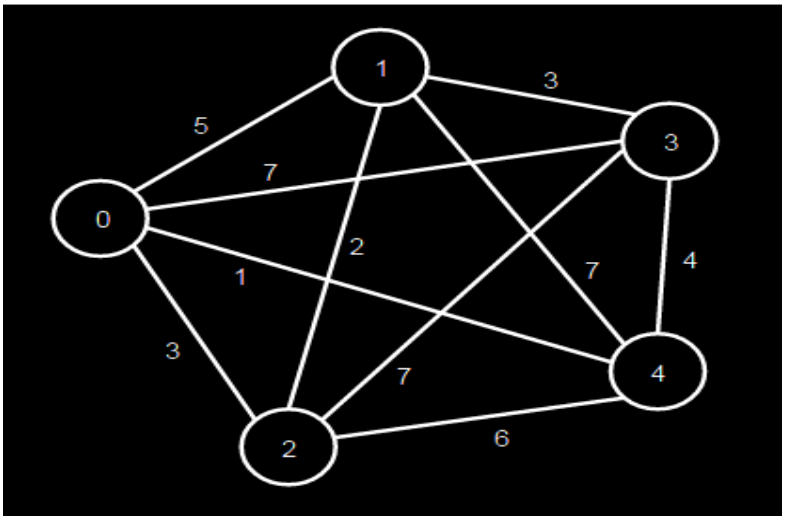

Figure 5.15: Graph3 – Mesh Topology

In the given Graph3, Source node is '1' and destination node is '2'.

I. Step1 is to take network input from the user.

```
Enter the number of nodes in the graph: 5
Enter the number of links in the graph: 10
Enter two node number of a link, forwhich you want to enter a value: 0 1
Enter a value for the provided link: 5
Enter two node number of a link, forwhich you want to enter a value: 0 2
Enter a value for the provided link: 3
Enter two node number of a link, forwhich you want to enter a value: 0 3
Enter a value for the provided link: 7
Enter two node number of a link, forwhich you want to enter a value: 0 4
Enter a value for the provided link: 1
Enter two node number of a link, forwhich you want to enter a value: 1 2
Enter a value for the provided link: 2
Enter two node number of a link, forwhich you want to enter a value: 1 3
Enter a value for the provided link: 3
Enter two node number of a link, forwhich you want to enter a value: 1 4
Enter a value for the provided link: 7
Enter two node number of a link, forwhich you want to enter a value: 2 3
Enter a value for the provided link: 7
Enter two node number of a link, forwhich you want to enter a value: 2 4
Enter a value for the provided link: 6
Enter two node number of a link, forwhich you want to enter a value: 3 4
Enter a value for the provided link: 4
Enter the source node: 1
Enter the destination node: 2
```

Figure 5.16: Network input for Graph3

Adjacency matrix for Graph3 is shown in Figure 5.17

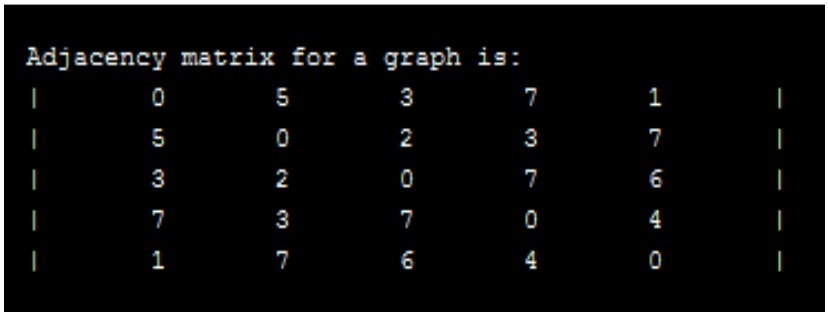

Figure 5.17: Adjacency matrix for Graph3

II. Next step is to find all possible paths from source node to the destination node.

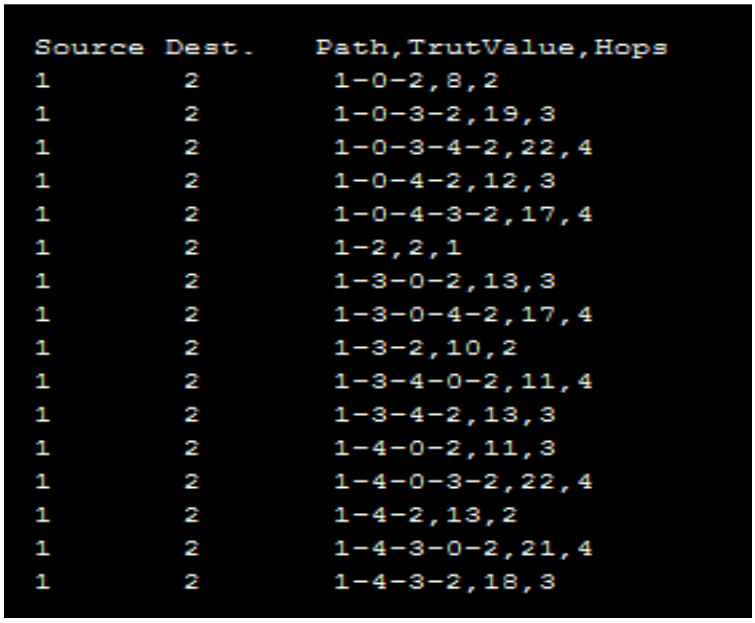

Figure 5.18: All possible paths in Graph3

III. Next step is to calculate the average trust value for each discovered path.

```
These are the final trust values corresponding to each path:

Avg Trust Value         No of hops      Path
    4.0                    2            1-0-2
    6.33                   3            1-0-3-2
    5.5                    4            1-0-3-4-2
    4.0                    3            1-0-4-2
    4.25                   4            1-0-4-3-2
    2.0                    1            1-2
    4.33                   3            1-3-0-2
    4.25                   4            1-3-0-4-2
    5.0                    2            1-3-2
    2.75                   4            1-3-4-0-2
    4.33                   3            1-3-4-2
    3.67                   3            1-4-0-2
    5.5                    4            1-4-0-3-2
    6.5                    2            1-4-2
    5.25                   4            1-4-3-0-2
    6.0                    3            1-4-3-2
```

Figure 5.19: Average trust value for each path in Graph3

IV. Next step is to calculate the maximum average trust value among all possible paths.

```
Trust Value
    6.5
```

Figure 5.20: Maximum average trust value in Graph3

V. Next step is to find the maximum average trust value with least number of hops.

```
Trust Value  No of hops
    6.5          2
```

Figure 5.21: Maximum trust value with the least number of hops in Graph3

VI. Next step is to find the most trusted path.

```
The most trusted path in this graph is:

Trust Value  No of hops   Trusted Path
    6.5          2          1-4-2
```

Figure 5.22: Most trusted path in Graph3

CHAPTER 6
CONCLUSION AND FUTUER SCOPE

6.1 Conclusion:

There are many challenges in MANET to design a routing protocol due to the resource constraints like bandwidth limited, limited battery life and so on. The mobility nature of MANET has been the main cause of changing topology frequently. Many applications of MANET have been developed and have its own nature. Some applications work on the sensitive information, for which it is necessary to provide security and provide a secured path from the source node to the destination node. Many different kinds of attacks are possible in MANET. So, security must be implemented in such networks. There are two important aspects in providing the security in MANET, either by using traditional security schemes such as cryptographic approaches or by using trust based schemes. Different kinds of secured routing protocols have been developed in both the aspects in providing the security. However, the main concern in routing protocols is to choose the secured path and to deliver the packets from the secured path. In the present era, researchers are taking more interest to provide the security by using concept of trust. In trust based schemes, different techniques have been used to calculate the trust among nodes in the network.

We proposed a Trust-based algorithm to find the most trusted path from the source node to a destination node in the network. We considered trust based model while developing the algorithm. Trust values between two nodes were assumed in the scale of 1 to 10. The proposed algorithm is basically based on the depth first traversal. The proposed algorithm traverses each node in the network and finds all possible paths from source node to destination node along with the total trust value and the number of hops required in each path; and then finally chooses the most trusted path which has the maximum trust value with the least number of hops, among the discovered path. The proposed algorithm is implemented for different topologies such as mesh topology.

6.2 Future Scope:

The proposed algorithm is a general approach to find the most trusted path from the source node to the destination node. The proposed algorithm is presently implemented in the Java programming language. For the future work, we can first calculate the trust values and implement the proposed algorithm with the trust based model in actual network. We can simulate it with any network simulator and can find the feasibility of the algorithm. Further, the algorithm implemented to find the trusted path can also be modified based upon certain applications.

REFERENCES

[1] Ghosekar, P., Katkar, G., & Ghorpade, P. (2010). Mobile ad hoc networking: imperatives and challenges. *IJCA Special Issue on MANETs*, *3*, 153-158.

[2] Macker, J., 1999. Mobile ad hoc networking (MANET): Routing protocol performance issues and evaluation considerations.

[3] Ahmed, A., Bakar, K.A., Channa, M.I., Haseeb, K. and Khan, A.W., 2015. A survey on trust based detection and isolation of malicious nodes in ad-hoc and sensor networks. *Frontiers of Computer Science*, *9*(2), pp.280-296.

[4] Hoebeke, J., Moerman, I., Dhoedt, B. and Demeester, P., 2004. An overview of mobile ad hoc networks: applications and challenges. *Journal-Communications Network*, *3*(3), pp.60-66.

[5] Cho, J.H., Swami, A. and Chen, I.R., 2011. A survey on trust management for mobile ad hoc networks. *Communications Surveys & Tutorials, IEEE,13*(4), pp.562-583.

[6] Marias, G.F., Georgiadis, P., Flitzanis, D. and Mandalas, K., 2006. Cooperation enforcement schemes for MANETs: A survey. *Wireless Communications and Mobile Computing*, *6*(3), pp.319-332.

[7] Govindan, K. and Mohapatra, P., 2012. Trust computations and trust dynamics in mobile adhoc networks: a survey. *Communications Surveys & Tutorials, IEEE*, *14*(2), pp.279-298.

[8] Sun, Y., Han, Z. and Liu, K.R., 2008. Defense of trust management vulnerabilities in distributed networks. *Communications Magazine, IEEE,46*(2), pp.112-119.

[9] URL:http://www.doc.ic.ac.uk/~ajs300/security/CIA.htm. Accessed: 2016-06-12. (Archived by WebCite® at http://www.webcitation.org/6iD42bxzt)

[10] Jasvinder, M.S., 2013. Effects of Black Hole Attack on an AODV Routing Protocol Through the Using Opnet Simulator. *International Journal of Advanced Research in Computer Science and Software Engineering*, *3*(8).

[11] Liu, Z., Joy, A.W. and Thompson, R.A., 2004, May. A dynamic trust model for mobile ad hoc networks. In *Distributed Computing Systems, 2004. FTDCS 2004. Proceedings. 10th IEEE International Workshop on Future Trends of* (pp. 80-85). IEEE.

[12] Sun, Y.L., Yu, W., Han, Z. and Liu, K.J., 2006. Information theoretic framework of trust modeling and evaluation for ad hoc networks. *Selected Areas in Communications, IEEE Journal on*, *24*(2), pp.305-317.

[13] Karlof, C. and Wagner, D., 2003. Secure routing in wireless sensor networks: Attacks and countermeasures. *Ad hoc networks*, *1*(2), pp.293-315.

[14] Wang, W., Bhargava, B., Lu, Y. and Wu, X., 2006. Defending against wormhole attacks in mobile ad hoc networks. *Wireless communications and mobile computing*, *6*(4), pp.483-503.

[15] Kannhavong, B., Nakayama, H., Nemoto, Y., Kato, N. and Jamalipour, A., 2007. A survey of routing attacks in mobile ad hoc networks. *Wireless communications, IEEE*, *14*(5), pp.85-91.

[16] Ameza, F., Assam, N. and Beghdad, R., 2010. Defending AODV routing protocol against the black hole attack. *International Journal of Computer Science and Information Security*, *8*(2).

[17] Li, J., Li, R. and Kato, J., 2008. Future trust management framework for mobile ad hoc networks. *Communications Magazine, IEEE*, *46*(4), pp.108-114.

[18] Bhalaji, N. and Shanmugam, A., 2009. Reliable Routing against selective packet drop attack in DSR based MANET. *Journal of Software*, *4*(6), pp.536-543.

[19] Cho, J.H., Swami, A. and Chen, R., 2012. Modeling and analysis of trust management with trust chain optimization in mobile ad hoc networks. *Journal of Network and Computer Applications*, *35*(3), pp.1001-1012.

[20] Pirzada, A.A. and McDonald, C., 2006. Trust establishment in pure ad-hoc networks. *Wireless Personal Communications*, *37*(1-2), pp.139-168.

[21] Shabut, A. et al., 2014. Recommendation Based Trust Model with an Effective

Defence Scheme for MANETs. IEEE Transactions on Mobile Computing, 1233(10), pp.2101–2115.

[22] Al-Karaki, J.N. and Kamal, A.E., 2008. Stimulating node cooperation in mobile ad hoc networks. *Wireless Personal Communications*, *44*(2), pp.219-239.

[23] Manikandan, S.P. and Manimegalai, R., 2013. Trust based routing to mitigate black hole attack in MANET. *Life Sci. J*, *10*(4), pp.490-498.

[24] Li, H. and Singhal, M., 2007. Trust management in distributed systems.*Computer*, (2), pp.45-53.

[25] Patel, V.H., Zaveri, M.A. and Rath, H.K., 2015. Trust Based Routing in Mobile Ad-Hoc Networks. *Lecture Notes on Software Engineering*, *3*(4), p.318.

[26] Khan, M.S., Midi, D., Khan, M.I. and Bertino, E., 2015, August. Adaptive Trust Threshold Strategy for Misbehaving Node Detection and Isolation. In*Trustcom/BigDataSE/ISPA, 2015 IEEE* (Vol. 1, pp. 718-725). IEEE.

[27] Kundu, J., Majumder, K. and De, D., 2015, February. Design and analysis of an efficient trust management scheme for clustered based MANET using Bayesian method. In *Electronics and Communication Systems (ICECS), 2015 2nd International Conference on* (pp. 60-65). IEEE.

[28] Saravanan, S., Chandrasekaran, R.M. and Nadu, T., 2015. Providing trust based key exchange security approach for intrusion detection using recommendation metrics over Manet (Skite) Stv. *Advances in Natural and Applied Sciences*, *9*(3), pp.71-80.

[29] Sridhar, S. and Baskaran, R., 2015. ANT Based Trustworthy Routing in Mobile Ad Hoc Networks Spotlighting Quality of Service. *American Journal of Computer Science and Information Technology (AJCSIT)*, *3*(1), pp.064-073.

[30] Chen, R. and Guo, J., 2015. Hierarchical trust management of community of interest groups in mobile ad hoc networks. *Ad Hoc Networks*, *33*, pp.154-167.

[31] Jain, Y.K. and Sharma, P., 2012, August. Trust based ad hoc on-demand distance vector for MANET. In *Proceedings of the National Conference on*

Security Issues in Network Technologies (NCSI-2012) (pp. 1-11).

[32] Babu, B.S. and Venkataram, P., 2011, July. A trust model for routing in MANETs: A cognitive agents based approach. In *Proceedings of the International Conference on Security and Management* (pp. 208-214).

[33] Jassim, H.S., Yussof, S., Kiong, T.S., Koh, S.P. and Ismail, R., 2009, December. A Routing Protocol based on Trusted and shortest Path selection for Mobile Ad hoc Network. In *Communications (MICC), 2009 IEEE 9th Malaysia International Conference on* (pp. 547-554). IEEE.

ANNEXTURE I

LIST OF ABBREVIATIONS

ACO	Ant Colony Optimization
AODV	Ad-hoc On-Demand Distance Vector
ATT	Adaptive Trust Threshold
BOB	Behavior-Observation-Belief
CH	Cluster Head
CIA	Confidentiality-Integrity-Availability
CM	Cluster Member
COI	Community of Interest
DoS	Denial of Service
DSR	Dynamic Source Routing
DTM-DSR	Dynamic Trust Mechanism-Dynamic Source Routing
GCS	Group Communication System
IDS	Intrusion Detection System
IT	Internal Trust
LIFO	Last in First out
MANET	Mobile Ad-hoc Network
NS	Network Simulator
OLSR	Optimized Link State Routing
OSI	Open System Interconnection
PAN	Personal Area Network
PKI	Public Key Infrastructure
QOS	Quality of Service
RREP	Route Reply
RREQ	Route Request
SGL	Subtask Group Leader
STV	Secure Trust Value
TCP	Transmission Control Protocol
TCS	Trust Correlation Service
TTL	Time to Live
VGA	Virtual Grid Architecture

ANNEXURE II

LIST OF PUBLICATIONS

Ankit Agrawal and Anil Kumar Verma, "**A review & impact of Trust Schemes in MANET**", in International Conference on Advances in Information Communication Technology & Computing (ACM) [AICTC -2016], Government Engineering College, Bikaner, India. [Accepted and Registered]

ANNEXURE III

VIDEO PRESENTATION

https://youtu.be/4bev1nmZ2Yc

ANNEXURE IV

PLAGIARISM REPORT

ANKIT AGRAWAL

ORIGINALITY REPORT

4%	0%	4%	0%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	Ahmed, Adnan, Kamalrulnizam Abu Bakar, Muhammad Ibrahim Channa, Khalid Haseeb, and Abdul Waheed Khan. "A survey on trust based detection and isolation of malicious nodes in ad-hoc and sensor networks", Frontiers of Computer Science, 2015. Publication	1%
2	Cho, Jin-Hee, Ananthram Swami, and Ing-Ray Chen. "A Survey on Trust Management for Mobile Ad Hoc Networks", IEEE Communications Surveys & Tutorials, 2011. Publication	<1%
3	Mallapur, Sujata V., and Siddarama R Patil. "Fuzzy Logic Based Trusted Candidate Selection for Stable Multipath Routing", International Journal of Information Technology and Computer Science, 2015. Publication	<1%
4	Jin-Hee Cho. "Modeling and Analysis of Trust Management for Cognitive Mission-Driven Group Communication Systems in Mobile Ad Hoc Networks", 2009 International Conference on Computational Science and	<1%