JIMMA UNIVERSITY

JIMMA INSTITUTE OF TECHNOLOGY

FACULTY OF COMPUTING AND INFORMATICS

Improving the Performance of AODV routing protocol to Mitigate Route Request flooding attack in MANET.

By:

Hirpa Girma

Advisor: Fisseha Bayu (PhD Candidate)

Co-advisor: Worku Birhane (MSc)

Thesis submitted to the Faculty Computing and Informatics in partial fulfillment of the requirements for the degree of Masters of Science in Computer Networking

September 8, 2021

JIMMA, ETHIOPIA

JIMMA UNIVERSITY

JIMMA INSTITUTE OF TECHNOLOGY

SCHOOL OF COMPUTING AND INFORMATICS

Improving the Performance of AODV routing protocol to Mitigate Route Request flooding attack in MANET

By:

Hirpa Girma

APPROVAL BY BOARD OF EXAMINERS

	Signature	**Date**
Mr. Kibebew Ababu (Ass. Professor) Head of Department	______________	___________
Mrs. Melishew Awoke (MSc) Internal Examiner	____________________	Sept. 01, 2021
Mr. Melkamu Deressa (PhD) External Examiner	___________________	Sept. 03, 2021
Mr. Bekan Kitaw (MSc) Chairperson	__________________	____________

DECLARATION

I, the undersigned, hereby declare that this thesis is my original work performed under the supervision of Fisseha Bayu (PhD Candidate) and Mr. Worku Birhanie, has not been presented as a thesis for a MSc degree program in any other university and all sources of materials used for the thesis are duly acknowledged.

Name: Hirpa Girma
Signature: ____________
Date of submission: ___________________
This thesis has been submitted for examination with my approval as university advisors.

1. Fisseha Bayu (PhD Candidate)
   Main-Advisor   Signature

2. Mr. Worku Birhanie
   Co-Advisor   Signature

ACKNOWLEDGMENT

First and foremost I praise the name of Almighty God, who gave me the strength, health and patience in every endeavor of my life. Second, I would like to express my genuine gratitude to my advisors, Fisseha Bayu (PhD candidate) and Mr. Worku Birhanie, for their guidance and advice during each stage of this research study. I want also to express my sincere appreciation to my friends for their encouragement and concern.

Finally, my heartfelt appreciation goes to my family for their love, support, patience, and encouragement throughout my life.

Table of Contents

LIST OF FIGURES

LIST OF TABLES

LIST OF ACRONYMS

AODV	Ad-hoc on-demand Distance Vector
AWK	Aho, Weinberger, Kernigham
CBR	Constant Bit Rate
DoS	Denial of Service
DSDV	Destination-Sequenced Distance Vector
DSR	Dynamic Source Routing
IEEE	Institute of Electrical and Electronics Engineers
MAC	Medium Access Control
MANET	Mobile Ad hoc Network
MRREQ	Mitigation of route request
NAM	Network Animator
NS-2	Network Simulator-2
OLSR	Optimized Link State Routing
OTCL	Object-oriented Tool Command Language
PDR	Packet Delivery Ratio
QoS	Quality of Service
RERR	Route Error
RREP	Route Reply

TCL	Tool Command Language
TCP	Transmission Control Protocol
TORA	Temporally Ordered Routing Algorithm

ABSTRACT

Mobile ad hoc Network (MANET) is a collection of network devices which are connected through the wireless links. Due to this, the attackers seek a chance to pierce into the network and become a cause of malfunctioning in the network. So, during the phases of routing in different types of protocols, each of the attack finds a way to degrade the performance of the Network.

In this research, Mitigation of route request flooding attack in AODV is designed and implemented on ns-2 simulator. To do this, we proposed new technique based on number of RREQ of nodes within **NET TRAVERSAL TIME in each nodes** and calculating the mean of RREQ by adding number of RREQ of each nodes and divide to number of nodes in the network finally add the RREQ RATE LIMIT of default AODV to identify threshold value to detect RREQ flooding attack node in AODV.

We have evaluated the performance of our proposed Mitigation of route request flooding attack by enhancing AODV (MRREQF-AODV) through MANET simulation environment. We have evaluated the performance of our proposed Within 20, 40 and 60 MANET nodes. The simulation experiment shows that the proposed **MRREQF-AODV** results within 20,40 and 60 MANET nodes are 93%,93% and 86% packet delivery ratio, 0.0165897,0.0184648 and 0.0098557 ms average end to end delay, 261.32, 266.1 and 246.93 kbps throughput, 2, 2.0125 and 1.9175 joule average residual energy respectively whereas **AIF-AODV** Within 20,40 and 60 MANET nodes are 17.9%,19.2% and 11.1% packet delivery ratio, 1.124,1.034 and 1.180 ms average end to end delay, 173.4,195 and 113.1 kbps throughput, and 1.680, 1.087 and 0.375 joule average residual energy respectively and in normal AODV within 20,40 and 60 MANET nodes are 16.46 %,16.35% and 24.41% packet delivery ratio, 1.3726, 1.361 and 1.28295 ms average end to end delay, 86.75, 39.12 and 107.61 kbps throughput, and 2.67, 0.6675 and 0.296667 joule average end to end delay, 173.4,195 and 113.1 kbps throughput, and 1.680, 1.087 and 0.375 joule average residual energy respectively. Thus, indicate that, MRREQF-AODV for detecting RREQ flooding attack Based on the proposed technique of AODV provides better performance than normal AODV and AIF-AODV and improves the service of mobile nodes.

Keywords: MANET, RREQ flooding attack, RREQ, NET TRAVERSAL TIME, Threshold

CHAPTER ONE: INTRODUCTION

1.1 Background

A Mobile ad hoc network is the group of wireless mobile computers (or nodes) in which nodes collaborate by forwarding packets for each other to allow them to communicate outside range of the direct wireless transmission [1]. Ad hoc networks require no centralized administration or fixed network infrastructure such as base stations or access points. MANET is an autonomous group of mobile users that communicate over reasonably slow wireless links. The following Figure 1.1 shows how MANET are connected with no centralized administration or fixed network infrastructure such as base stations or access points.

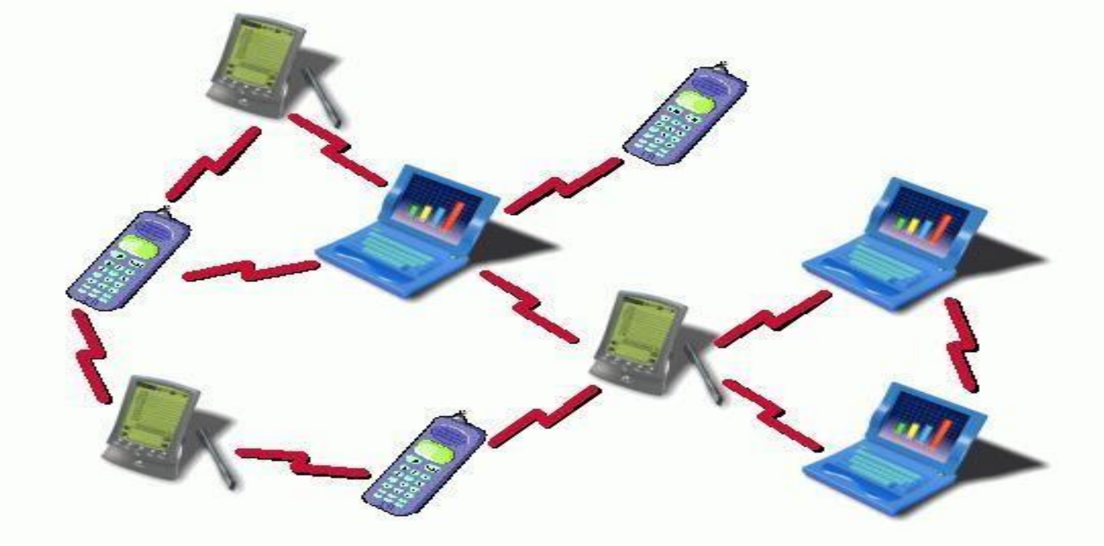


Figure 1. 1 MANET structure

MANETs are vulnerable to attacker as compared to wired networks due to mobile nodes. Due to these vulnerabilities, MANETs are more prone to malicious nodes. The Ad-hoc networks have various vulnerabilities like scalability, dynamic topology and infrastructure less networks, lack of centralized node, limited resources, and bandwidth constrained.

As the network is wireless, security becomes the major issue in Mobile Ad hoc Networks. Some of the attacks such as black hole attack, flooding attack, and Wormhole Attack are due to misbehavior of malicious nodes, which disrupts the transmission. Most of the reactive protocols like AODV are prone to flooding attacks which is a kind of denial of service attacks during their route discovery process. A malicious node may actively involve in the flooding attack by repeatedly sending RREQ or garbage data packets to different destinations some of which never exists. A neighboring victim node may drain its resources like battery power, consuming bandwidth, processing time by involving itself in the routing traffic.

Out of various attacks in MANETs, Flooding attack is the most hazardous attack, which is responsible for reducing the network performance by consuming network resources. This attack is very easy to implement in the network, but it is a most hazardous attack. This type of attack can be implemented by using an excess of route requests or by flooding large amount of data in the network. In this, malicious nodes flood excess of fake route requests in the network to decrease the performance of the network. In Flooding attack, the malicious nodes get into the network and set various paths with different nodes in the network. After establishing different paths in the network, these malicious nodes inject large amounts of RREQs packets for getting paths to different destination nodes. These large amounts of useless data packets congest the network. Due to this, the number of nodes other than the malicious nodes will be busy all of the time while receiving unwanted and useless data packets. The main aim of the rreq flooding attack is to consume and exhaust the network resources. The routing operation is disrupted to a large extent by this attack. The rreq flooding attack is used to degrade the performance of the network, so this attack is most hazardous attack.

Therefore routing protocols in MANET should be provided with algorithms and techniques that minimizing and identifying these attacks in order to preserve these properties and finally improves the performance of AODV routing protocol by improving network metrics[2].

As an example, RREQs flooding attack is shown in the Figure1.2

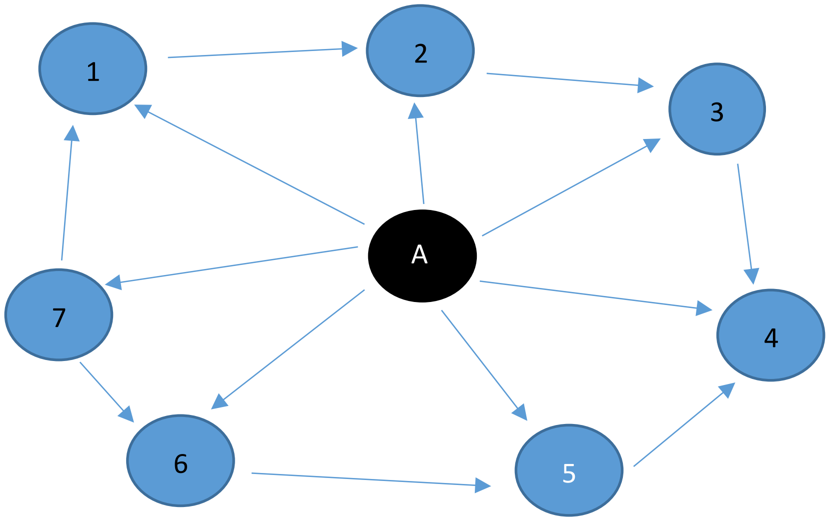


Figure 1. 2 Overview of RREQ Flooding Attack by node A in AODV

● A Attacker node

→ RREQ flooding path

Because of the above-mentioned properties, the nodes and data in MANET are vulnerable to a variety of threats and attacks [2], [3]. To overcome the above problem, we propose new algorithm based on number of RREQ generated by each nodes with in NET TRAVERSAL TIME and calculating the mean of RREQ plus route request rate limit of default AODV to identify threshold value to detect the attacker node in AODV. This new prevention algorithm promises to resolve the problems of the existing AODV routing protocol by considering multiple constraints like: network overheads even in large networks, network performance like: drop packets, packet delivery ratio, end to end delay, residual energy and throughput.

1.2. Statement of the Problem

Attackers can exploit on-demand routing protocols to initiate different attacks. As described in [4], in AODV flooding attack, the attacker either broadcasts a lot of Route Request (RREQ) packets for a node which is not in the network or sends a lot of attacking data packets to exhaust the communication [5] bandwidth and nodes' resources. The RREQ messages will be received by all the neighbors of the originator since RREQ packets are broadcasted to discover a route.

The intermediate nodes will look up their routing tables for possible route to the destination, but none of the nodes will have a route to the destination node specified in the RREQ message and the intermediate nodes will flood the RREQ packet to their respective neighbors. The malicious node may worsen the situation by repeating this procedure frequently. As a result, the requests will be routed through all nodes there by creating a congested network and consuming the scarce network resources like bandwidth and battery life. By doing so, the route request flooding attack may lead to denial of service attack as there will be more RREQ packets in the network and the nodes will be busy in processing and forwarding these fake requests rather than relaying genuine packets.

This is due to the fact that control packets like RREQ have higher priority than data packets. The attacker can also send useless data packets to overuse the available bandwidth.

1.3 Motivation

As the network is wireless, security becomes the major issue in Mobile Ad hoc Networks. Some of the attacks such as flooding attack, Blackhole attack, gray hole and wormhole attacks are due to misbehavior of malicious nodes, which disrupts the transmission. Most of the reactive protocols like AODV are prone to route request flooding attacks which is a kind of denial of service attacks during their route discovery process. A malicious node may actively involve in the route request flooding attack by repeatedly sending RREQ or garbage data packets to different destinations some of which never exists. A neighboring victim node may drain its resources like battery power, consuming bandwidth, processing time by involving itself in the routing traffic. The idea is motivated to study in the area of mitigation of route request flooding attack in AODV. Here, the prime motivation of our thesis is to mitigate and identify route request flooding attack in AODV to increase the performance of AODV routing protocol in MANET[6]. In the above section (1.2), we have explained the limitations of our current AODV routing protocols in MANET.

1.4. Research Questions

The questions that addressed in this research are listed in the following way.

1. Why AODV is more vulnerable for RREQ flooding attacks?
2. What are the limitations of previous researchers?

3. How to identify and mitigate RREQ flooding attacks over AODV?
4. Which software tool is preferable for this research?

The above listed research question will be answered on the progress of this research.

1.5. Solution proposed

To overcome the above problem, we propose new technique depend on number of RREQ generated by each nodes with in *NET TRAVERSAL TIME* and calculating the mean of RREQ plus route request rate limit of default AODV to identify threshold value to detect the attacker node in AODV. According to AODV RFC 3561 [6] protocols standard rule, a normal nodes can a request 10 times per second to transmit data. According to AODV RFC 3561 rule, a node *should not* originate more than RREQ_RATELIMIT RREQ messages per second. After broadcasting a RREQ, a node waits for a RREP (or other control message with current information regarding a route to the appropriate destination). If a route is not received within NET_TRAVERSAL_TIME milliseconds, the node MAY try again to discover a route by broadcasting another RREQ, up to a maximum of RREQ_RETRIES times at the maximum TTL value. In our technique, first each node in the network has a table called Request Counter that records how much number of RREQ generated in each nodes with in NET TRAVERSAL TIME. Finally, mean of RREQ with in this time interval and add RREQ RATE LIMIT to it to determine the threshold value to detect RREQ flooding attack. So, whenever a node receives a route request greater than this threshold value, the request is dropped and add to the blacklist table. However, when requesting nodes make a request less than threshold, the nodes processes the request normally. This new prevention algorithm promises to resolve the problems of the existing AODV routing protocol by considering multiple constraints like: network overheads even in large networks, network performance like: drop packets, packet sent, packet delivery ratio, end to end delay and throughput.

1.6. Objectives

1.6.1 General objective

The general objective of this thesis is to develop the mechanism to mitigate RREQ flooding attack by modifying AODV routing protocol in MANET.

1.6.2 Specific objectives

To achieve the above general objective, it is essential to go through the following basic tasks.

- ✓ To identify and investigate the problems of RREQ flooding attack in MANETs.
- ✓ To increase the performance of AODV.
- ✓ To implement the mechanism of minimizing and identifying RREQ flooding attack by modifying AODV routing protocol on simulation environment.
- ✓ To testing and evaluating the performance of the new algorithms through simulations.
- ✓ Compare the performance with existing normal AODV and AIF AODV protocol.

1.7 Methods

The activities to be carried out through the research to accomplish those objectives are as follows.

Literature Review

Exhaustive study and explorations would made on the areas related to prevention of RREQ flooding attack mechanism of AODV in MANETs. This is accomplished by reading different books, journals or conference papers which have been done so far with different approaches, so as to have sufficient understanding of the problem. Techniques and approaches appropriate for development of preventing RREQ flooding attack algorithm for AODV routing protocol. After deep understood of the problem we have proposed a new technique.

Design

In the design phase, proposed models and algorithms which are specified in the objective of this Paper have been designed.

Implementation

Due to high cost of MANET nodes, we have been implemented the proposed that minimize and identify RREQ flooding attack in AODV. We have used a simulated MANET environment.

Experimental Evaluation

Experiments have been conducted to test the effectiveness of our proposed of minimize and identify RREQ flooding attack in AODV routing protocol in MANETs, and the performance evaluation of this work have been carried out in comparison to evaluate in terms of its objective and contributions in comparison to what is already done so far which is existing standard AODV routing protocol using MANET simulation environment. Evaluation have been conducted by considering different QoS metrics.

1.8 Scope and Limitations of the study

The scope of this research is limited to:

- ✓ Designing and implementing prevention of RREQ flooding attack in MANETs using AODV protocols.
- ✓ Identify the available attacker nodes during communication from source to destination on each path.
- ✓ The study will take attention on providing lower end to end packet delivery for application that needs QoS routing by addressing issues like packet delivery ratio, residual energy, e2end delay and throughput.

Our proposed work will not cover the following tasks:

- ✓ Protecting of others protocols from attacker except against AODV.
- ✓ Protecting of other types of attacks except RREQ flooding attack.

1.9 Significance of the study

Security is a vital scope in MANET to protect communication between mobile nodes [7]. Ad-hoc

On-demand Distance Vector (AODV) is one of the on-demand reactive routing protocols in MANET that initially was improved without considering security protection. So, the significant of this paper have been secured and detected attackers in AODV routing protocol in MANET. In the present study, after reviewing secured protocols of some previous researches, an improved protocol is proposed to enhance the security of AODV routing protocol against RREQ flooding attack. For this purpose, we used a different techniques like: number of rreq generated by each nodes with in NET TRAVERSAL TIME and determine average of rreq (mean of rreq) in order to know threshold value plus RREQ RATE LIMIT of normal AODV to detect the compromised nodes and malicious behaviors inside MANET; which leads to the low delay and high performance in the network.

Generally, the significance of this study are:

- ✓ To avoid congesting the network with fake request packets in order to eliminate their bad effects on the network.
- ✓ To decide whether the request is received from an attacker node or from a normal node.
- ✓ To reduce nodes power consumption during their communication.
- ✓ To increase performance in the network.

1.10 Thesis organization

The rest of the document is organized as follows. **Chapter two discusses about literature review**, in this chapter, routing protocols in MANETs are reviewed and specifically the AODV routing protocol is studied in detail regarding its operation including route discovery and route maintenance. Also vulnerabilities of MANETs: The sources of vulnerabilities and the types of attacks in MANETs including the RREQ flooding attack are also studied in this section. **Chapter three** discusses the works related to Prevention of flooding attack in MANETs. **Chapter four** introduces the NS-2 simulator and its installation, implementation of AODV in NS-2, modeling flooding attack and their implementations. The simulation study is discussed in **chapter five**. Here, the simulation scenarios used in this study, the performance metrics, the results of the simulations and the discussions of the results are presented. **Chapter six** presents the conclusions, contribution and future work. Then the reference materials used to develop this thesis work are presented next. Finally, under the appendix: sample of TCL codes, AWK scripts and C++ codes of the flooding attack plus the prevention mechanism were attached.

CHAPTER TWO: LITERATURE REVIEW

2. Overview of routing protocols

This chapter discusses the overview of routing protocols and its security in MANETs and the working principle of AODV routing protocol.

2.1. Routing protocols

Routing is the process of selecting a path for traffic in a network or between or across multiple networks. Routing protocols in mobile ad hoc network means that the mobile nodes will search for a route or path to connect to each other and share the data packets. Protocols are the set of rules through which two or more devices (mobile nodes, computers or electronic device) can communicate to each other. In mobile ad hoc networks the routing is mostly done with the help of routing tables. These tables are kept in the memory cache of these mobile nodes. When routing process is going on, it route the data packets in different mechanisms. The first is unicast, in which the source directly sends the data packets to the destination. The second is broadcast; it means the source node sends messages to all the near and far nodes in the network. The third is any cast, in this the source node sends data packet to anyone which is not in the node group [8].

In adhoc wireless networks, there are three types of routing protocols in MANETs. Those are: reactive (on demand), proactive (table-driven) and hybrid routing protocols [9] Figure 2.1 shows the three types of ad hoc routing protocols and some list of the available routing protocols for that category.

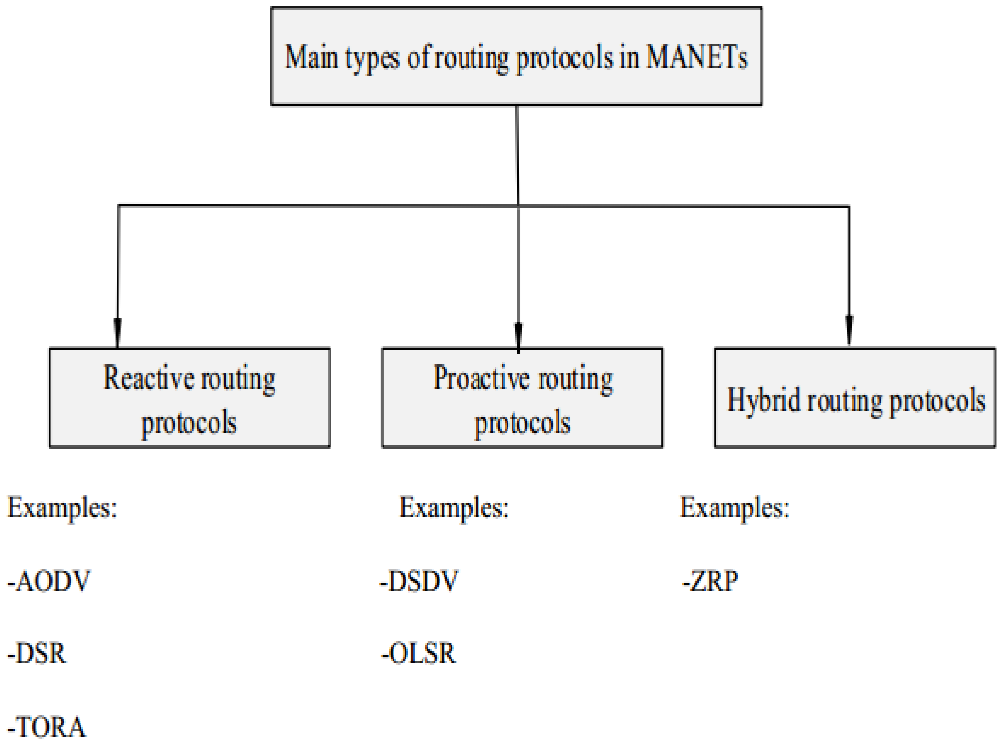

Main types of routing protocols in MANETs
Reactive routing protocols
Proactive routing protocols
Hybrid routing protocols
Examples:
-AODV
-DSR
-TORA
Examples:
-DSDV
-OLSR
Examples:
-ZRP

Figure 2. 1 Routing protocols in MANETs

Reactive protocols are also known as on-demand driven reactive protocols. These protocols do not start route discovery by themselves, until it has been made a request to, when a source node asks for finding a route. That is why these techniques are known as reactive techniques. These protocols install routes when required. Examples of reactive MANET protocols include Ad Hoc On-Demand Distance Vector (AODV), Dynamic Source Routing (DSR), and Temporally Ordered Routing Algorithm (TORA) [8], [9].

Proactive protocols are also table-driven routing protocols that try to keep a record of fresh and updated network routes. All the nodes in the network have a table to store the routing information. The nodes exchange topology information so that they can all have the same view of the network. The exchanged information helps to reflect any changes in the topology. Whenever a node needs to send messages, it just searches the routing table for the path to the destination. The sending of the message is not delayed by the remote route discovery. Maintaining an up-to-date topology in the routing tables causes a high control overhead.

Examples of proactive MANET protocols include Optimized Link State Routing (OLSR) and Destination-Sequenced Distance Vector (DSDV) [8], [10].

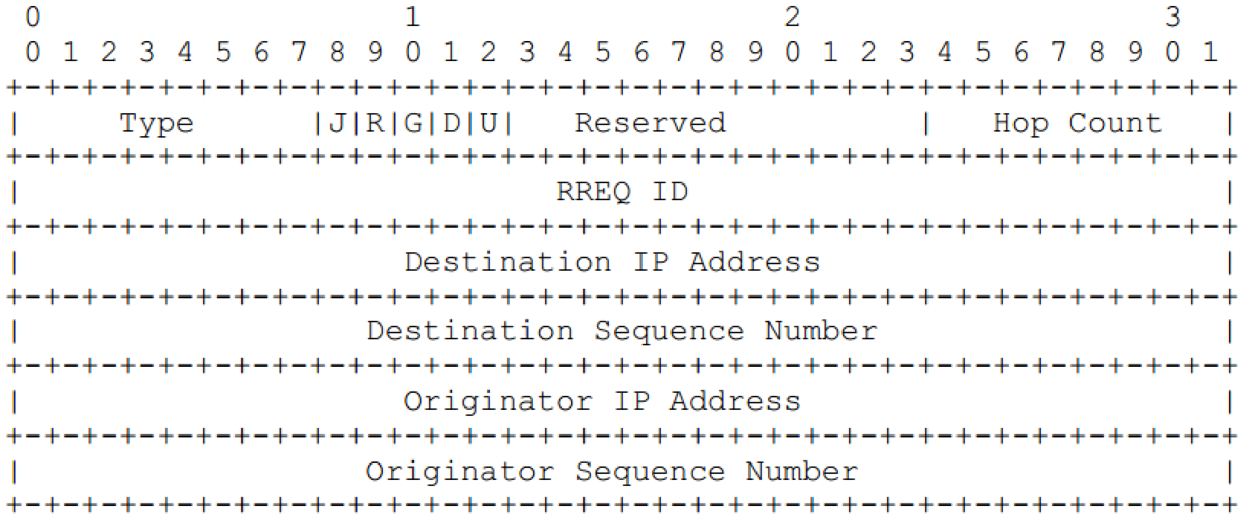
Hybrid routing protocols such as Zone Routing Protocol (ZRP) and Secured Link State Protocol (SLSP) combine the best features of both reactive and proactive routing protocols. For example, a node communicates with its neighbors using a proactive routing protocol, and uses a reactive protocol to communicate with nodes farther away.

2.2. Why AODV is selected for our thesis?

The main reason to select AODV routing protocol is that: As the demand for the MANETs are increasing day to day, due to the increasing demand for MANETs in various areas such as in Military operations and in flood affected areas, threat of security has also increased [10]. The main challenge in MANET is to design the strong security solution that can protect MANET from various routing attacks. From those routing attack RREQ flooding attack is one. RREQ Flooding attack launched at network layer is a kind of Denial of service (DOS) attack which is distributive in nature and can exhaust the victim’s network resources such as bandwidth, energy, computing power etc. The route discovery scheme in reactive routing protocols like Ad hoc On Demand Distance Vector (AODV) used in MANET makes it easier for malicious nodes to launch connection request floods by flooding the route request packets (RREQ) on the network. Due to the above reasons, we have selected this protocol to improve its performance by detecting RREQ flooding attack.

2.3. Ad hoc on-demand distance vector (AODV)

The Ad hoc On-Demand Distance Vector (AODV) algorithm enables dynamic, self-starting, multihop routing between participating mobile nodes wishing to establish and maintain an ad hoc network [11]. AODV allows mobile nodes to obtain routes quickly for new destinations, and does not require nodes to maintain routes to destinations that are not in active communication. AODV allows mobile nodes to respond to link breakages and changes in network topology in a timely manner. One distinguishing feature of AODV is its use of a destination sequence number for each route entry. The destination sequence number is created by the destination to be included along with any route information it sends to requesting nodes. Using destination sequence numbers ensures loop freedom and is simple to program. Given the choice between two routes to a destination, a requesting node is required to select the one with the greatest sequence number. Route Requests (RREQs), Route Replies (RREPs), and Route Errors (RERRs) are the message types defined by AODV. These message types are received via UDP, and normal IP header processing applies. The message formats for RREQ, RREP and RERR messages are defined in the draft of AODV, RFC 3561. The message format for RREQ includes the following fields.



```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|     Type      |J|R|G|D|U|   Reserved          |   Hop Count   |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                            RREQ ID                            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                    Destination IP Address                     |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                  Destination Sequence Number                  |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                    Originator IP Address                      |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                  Originator Sequence Number                   |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 2. 2 Route Request (RREQ) Message Format in AODV [11]

RREQ ID: A sequence number uniquely identifying the particular RREQ when taken in conjunction with the originating node's IP address.

Destination IP Address: The IP address of the destination for which a route is desired.

Destination Sequence: Number The latest sequence number received in the past by the originator for any route towards the destination.

Originator (source) IP Address: The IP address of the node which originated the Route Request.

Originator Sequence Number: The current sequence number to be used in the route entry pointing towards the originator of the route request.

Reserved: Sent as 0; ignored on reception.

Hop Count: The number of hops from the Originator IP Address to the node handling the request.

The Route Reply (RREP) message format also includes Destination IP Address, which is the IP Address of the destination for which a route is supplied, Destination Sequence Number, which is the destination sequence number associated to the route, and Originator IP Address is the IP address of the node which originated the RREQ for which the route is supplied. AODV uses RREQ and RREP to find routes for a destination and RERR is used for route maintenance during failure. Route table information must be kept even for short-lived routes. Some of the fields AODV uses with each route table entry include Destination IP Address, Destination Sequence Number, Network Interface, Hop Count (number of hops needed to reach destination), Next Hop, Lifetime (expiration or deletion time of the route) and other state and routing flags (e.g., valid, invalid, repairable, being repaired).

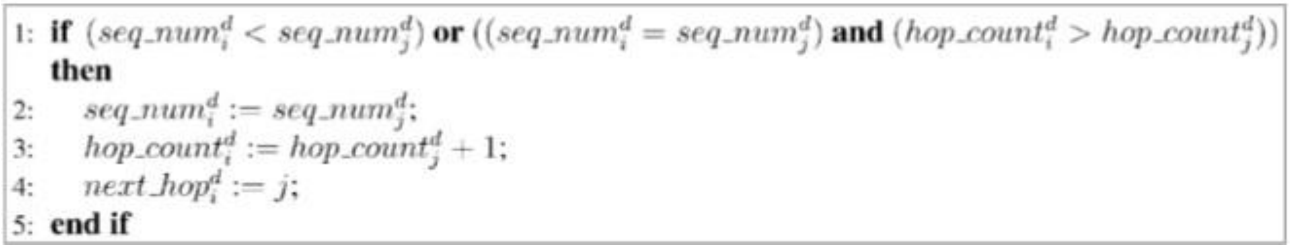
2.4 AODV OPERATION

As described in [12], [13] every route table entry at every node must include the latest information available about the sequence number for the IP address of the destination node for which the route table entry is maintained. This sequence number is called the "destination sequence number". It is updated whenever a node receives new (i.e., not stale) information about the sequence number from RREQ, RREP, or RERR messages that may be received related to that destination. AODV depends on each node in the network to own and maintain its destination sequence number to guarantee the loop freedom of all routes towards that node.

2.4.1. UPDATING ROUTING TABLE IN AODV

When a node receives an AODV control packet from a neighbor, it checks its route table for an entry for the destination. In the event that there is no corresponding entry for that destination, an entry is created. The sequence number is either determined from the information contained in the control packet, or else the valid sequence number field is set to false [14]. The route is only updated according to the following rule.

A node i applies the rules shown in figure 3.3 whenever it receives a route advertisement for destination d from a neighbor j. The variables seqnum^{d}i, hopcount^{d}i, and nexthop^{d}i denote the destination sequence number, the hop count and the next hop, respectively, for destination d at node i and seqnum^{d}j, hopcount^{d}j, and nexthop^{d}j denote the destination sequence number, the hop count and the next hop, respectively, for destination d at node j. This means that an entry will be updated if the new information has greater or equal sequence number than the receiving nodes has [14].



```
if $(seq\_num_i^d < seq\_num_j^d)$ or $((seq\_num_i^d = seq\_num_j^d)$ and $(hop\_count_i^d > hop\_count_j^d))$ then
    $seq\_num_i^d := seq\_num_j^d;$
    $hop\_count_i^d := hop\_count_j^d + 1;$
    $next\_hop_i^d := j;$
end if
```

Figure 2. 3 AODV route update rules [14]

For each valid route maintained by a node as a routing table entry, the node also maintains a list of precursors that may be forwarding packets on this route. These precursors will receive notifications from the node in the event of detection of the loss of the next hop link. The list of precursors in a routing table entry contains those neighboring nodes to which a route reply was generated or forwarded [14]. Whenever a node wants to send data to a given destination node and does not have any route to that node in its route table, it initiates a process called a route discovery procedure which is discussed in the following subtopic.

2.4.2. ROUTE DISCOVERY PROCEDURE IN AODV

IN AODV Route discovery [15], [16], [17] begins when a source node needs a route to a destination and does not have one available in its routing table. It first places the destination IP address and last known sequence number for that destination, as well as its own IP address and current sequence number, into a Route Request (RREQ) message. The RREQ ID field is incremented by one from the last RREQ ID used by the current node. The Hop Count field is set to zero. Before broadcasting the RREQ, the originating node buffers the RREQ ID and the Originator IP address (its own address) of the RREQ for PATH_DISCOVERY_TIME. In this way, when the node receives the packet again from its neighbors, it will not reprocess and re-forward the packet. A node should not originate more than RREQ_RATELIMIT RREQ messages per second, which in this case is set to 10. After broadcasting a RREQ, a node waits for a RREP. If a reply is not received within NET_TRAVERSAL_TIME milliseconds, the node try again to discover a route by broadcasting another RREQ, up to a maximum of

RREQ_RETRIES times at the maximum TTL value. Each new attempt must increment and update the RREQ ID.

When a node receives the RREQ, it first determines whether it has received a RREQ with the same Originator IP Address and RREQ ID within at least the last PATH_DISCOVERY_TIME. If such a RREQ has been received, the node silently discards the newly received RREQ. If not, it creates a reverse route entry for the source node in its route table. It then checks whether it has a fresh enough route to the destination node.

In order to respond to the RREQ, the node must either be the destination itself, or intermediate node with an unexpired route to the destination whose corresponding sequence number is at least as great as that contained in the RREQ. If neither of these conditions is met, the node rebroadcasts the RREQ to its respective neighbors by incrementing the hop count value in the RREQ by one, to account for the new hop through the intermediate node. Either the destination itself or an intermediate node with a fresh enough route will respond for the route request message using a Route Reply (RREP) message. Once created, the RREP is unicast to the next hop toward the originator of the RREQ as indicated by the route table entry for that originator. As the RREP is forwarded back towards the node which originated the RREQ message, the Hop Count field is incremented by one at each hop. Thus, when the RREP reaches the originator, the Hop Count represents the distance, in hops, of the destination from the originator.

When an intermediate node receives the RREP, it creates a forward route entry for the destination node in its route table, and then forwards the RREP to the source node. If the current node is not the node indicated by the Originator IP Address in the RREP message and a forward route has been created or updated, the node consults its route table entry for the originating node to determine the next hop for the RREP packet, and then forwards the RREP towards the originator using the information in that route table entry. Once the source node receives the RREP, it can begin using the route to transmit data packets to the destination. If it later receives a RREP with a greater destination sequence number or equivalent sequence number and smaller hop count, it updates its route table entry and begins using the new route.

The route discovery procedure can be explained diagrammatically as shown in figure 2.4 The diagram is taken from [16], [17] with modification to include the reverse and forward routes setup. If node 1 wants to send data to a destination 7 and does not have valid route to 7, it floods

RREQ messages (lines seen in solid red color) to its neighbors 2 and 3). The intermediate nodes then cache the received message and flood the request to their respective neighbors if they don't have fresh enough route to 7. When the RREQ reaches 7, node 7 prepares a RREP (lines seen in dotted red color) and this reply is unicasted to the originator using the partial route established during the propagation of RREQ messages. The intermediate nodes then forward the RREP to the originator by adding forward path to their table. When node A receives a reply, it immediately starts to forward data to 7 using the established route.

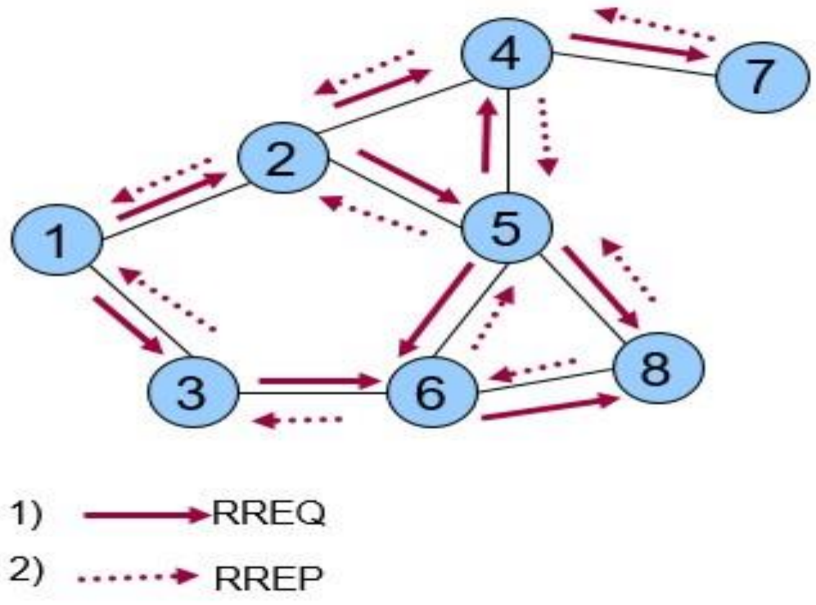


Figure 2. 4 Route discovery procedure in AODV [16], [17].

2.4.3 ROUTE MAINTENANCE IN AODV

On-Demand protocols also employ a route maintenance procedure [16], [18] where nodes monitor the operation of the route and inform the sender of any routing error. A node may offer connectivity information by broadcasting local Hello messages. A node should only use hello messages if it is part of an active route. Every HELLO_INTERVAL milliseconds, the node checks whether it has sent a broadcast (e.g., a RREQ or an appropriate layer 2 message) within the last HELLO_INTERVAL. If it has not, it may broadcast a RREP with TTL = 1, called a Hello message. Local Hello messages are used to determine local connectivity, which can reduce response time to routing requests and can trigger updates when necessary. Link breaks in non-active links do not trigger any protocol action. If so, it creates a Route Error (RERR) packet and the RERR contains the IP address of each destination that is now unreachable due to the link break. The RERR also contains the sequence number of each such destination, incremented by one. The node then broadcasts the packet and invalidates those routes in its route table. When a neighboring node receives the RERR, it a next hop. If one or more routes are deleted, the node then goes through the same process, whereby it checks whether any of its neighbors route through it to reach the destinations. If so, it creates and broadcasts its own RERR message.

A node initiates processing for a RERR message in three situations:

1. If it detects a link break for the next hop of an active route in its routing table while transmitting data (and route repair, if attempted, was unsuccessful), or
2. If it gets a data packet destined to a node for which it does not have an active route and is not repairing (if using local repair), or
3. If it receives a RERR from a neighbor for one or more active routes.

For case (1), the node first makes a list of unreachable destinations consisting of the unreachable neighbor and any additional destinations in the local routing table that use the unreachable neighbor as the next hop. For case (2), there is only one unreachable destination, which is the destination of the data packet that cannot be delivered. For case (3), the list should consist of those destinations in the RERR for which there exists a corresponding entry in the local routing table that has the transmitter of the received RERR as the next hop.

No	**Message types**	**Purpose**
1	RREQ	Used to find routes and is initiated by a source node.
2	RREP	Response to RREQ message
3	RERR	Notifies link failures
4	HELLO	Provides connectivity information

Table 2. 1 Message types of AODV

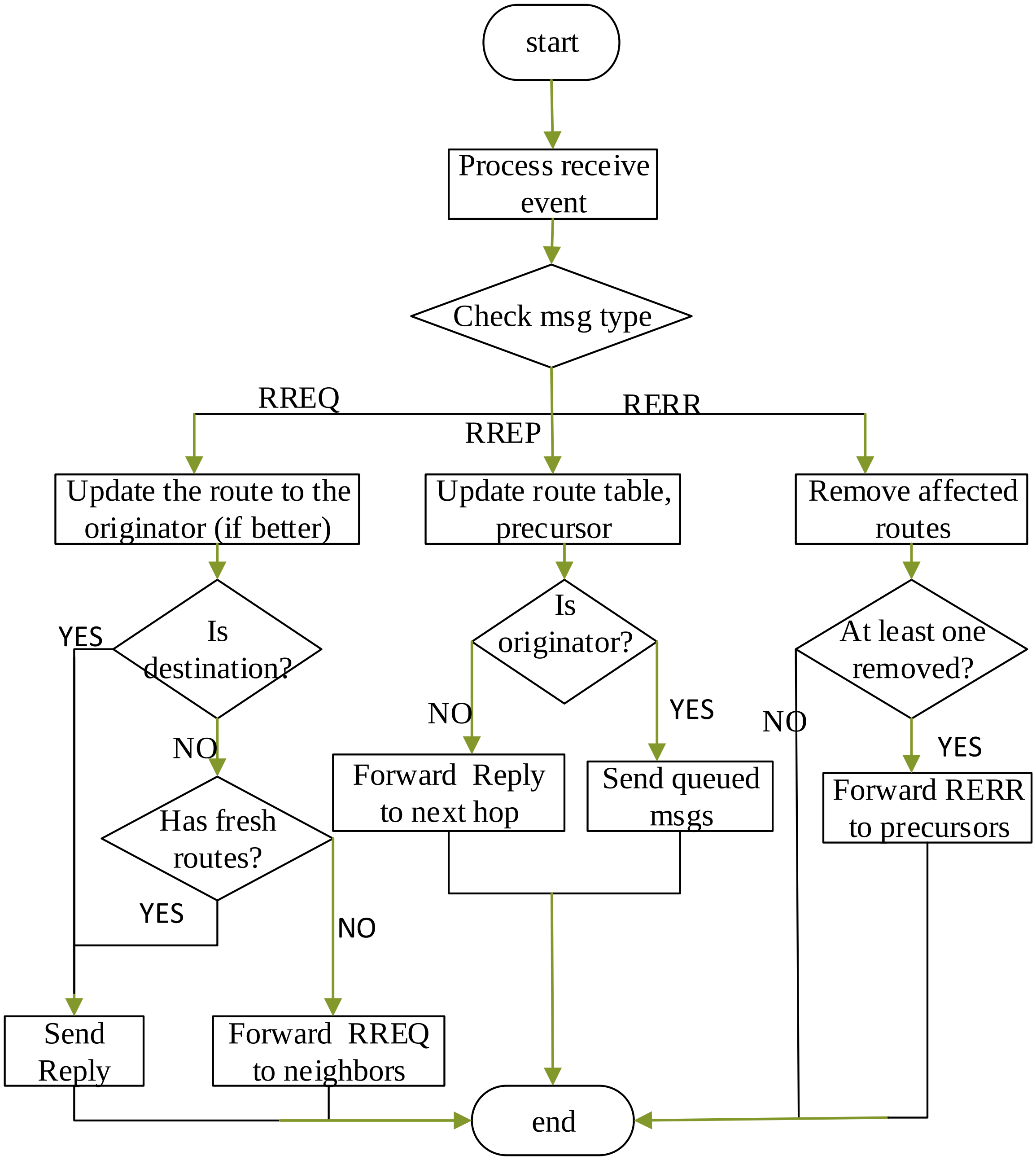


Figure 2. 5 Processing an incoming message in AODV [18]

The above flow chart shows that the overall of an AODV protocol when a node is processing an incoming AODV message. Both RREQ and RREP are processed during route discovery procedure and RRER is generated and processed when an active link fails.

2.5 MANET SECURITY

A MANET is a distributed infrastructure less network and mainly relies on individual security solutions from each mobile node and therefore centralized security control is hard to implement in it. The nature of ad hoc networks makes them vulnerable to various forms of attack. The random nature of these networks makes enforcement of security a challenging issue. This chapter presents the main vulnerabilities in the mobile ad hoc networks, which have made it much easier to suffer from attacks. Then it presents the main attack types that exist in it and effects of security in MANETs.

Security in Mobile Ad-Hoc Network (MANET) is the most important concern for the basic functionality of network. Availability of network services, confidentiality and integrity of the data can be achieved by assuring that security issues have been met. MANET often suffer from several security attacks because of its features like open medium, changing its topology dynamically, cooperative algorithms, , lack of central monitoring and management and no clear defense mechanism.

Moreover the newly requested route has the feasibility of possessing a few malicious nodes, resulting in failure of new route too. The fundamental issue with frequently used routing protocols is that they rely on all mobile nodes of network and based on the assumption that nodes will behave or cooperate properly but there might be a condition where some nodes are not behaving properly. Majority of the ad hoc network routing protocols become inefficient and show reduced performance while mitigating with big number of misbehaving nodes [15].

From those routing protocols, AODV is designed for use in networks where the nodes can all trust each other, either by use of Pre-configured keys, or because it is known that there are no malicious intruder nodes. AODV has been designed to reduce the dissemination of control traffic and eliminate overhead on data traffic, in order to improve scalability and performance.

2.5.1. Vulnerability of MANETs

MANET is more vulnerable than wired network due to mobile nodes, threats from compromised nodes inside the network, limited physical security, dynamic topology, scalability and lack of centralized management. Because of these vulnerabilities, MANET is more prone to malicious attacks. Some of the vulnerabilities discussed in [19], [20] are as follows:-

Lack of centralized management: MANET doesn’t have a centralized monitor server. The absence of management makes the detection of attacks difficult because it is not east to monitor the traffic in a highly dynamic and large scale ad-hoc network. Lack of centralized management will impede trust management for nodes.

Resource availability: Resource availability is a major issue in MANET. Providing secure communication in such changing environment as well as protection against specific threats and attacks, leads to development of various security schemes and architectures. Collaborative ad-hoc environments also allow implementation of self-organized security mechanism.

Scalability: Due to mobility of nodes, scale of ad-hoc network changing all the time. So scalability is a major issue concerning security. Security mechanism should be capable of handling a large network as well as small ones.

Dynamic topology: Dynamic topology and changeable nodes membership may disturb the trust relationship among nodes. The trust may also be disturbed if some nodes are detected as compromised. This dynamic behavior could be better protected with distributed and adaptive security mechanisms.

Bandwidth constraint: Variable low capacity links exists as compared to wireless network which are more susceptible to external noise, interference and signal attenuation effects.

Adversary inside the Network: The mobile nodes within the MANET can freely join and leave the network. The nodes within network may also behave maliciously. This is hard to detect that the behavior of the node is malicious.

2.5.2 Attacks in MANET

Securing wireless ad-hoc networks is a highly challenging issue. Understanding possible form of attacks is always the first step towards developing good security solutions. Security of communication in MANET is important for secure transmission of information.

Absence of any central co-ordination mechanism and shared wireless medium makes MANET more vulnerable to digital/cyber-attacks than wired network. There are a number of attacks that affect MANET [21]. These attacks can be classified into two types:

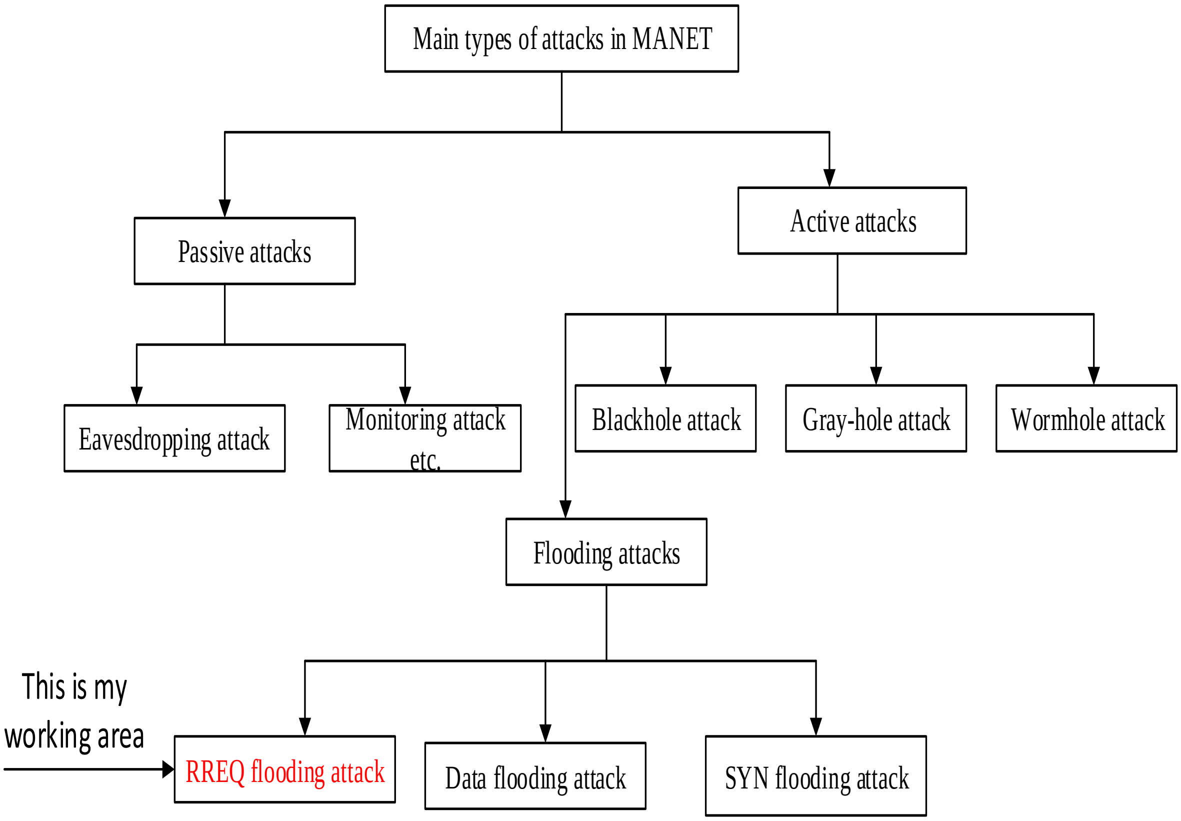


Figure 2. 6 Types of attacks over view

1. **Passive Attacks**

A passive attack does not disrupt proper operation of the network. The attacker snoops the data exchanged in the network without altering it. Here, the requirement of confidentiality can be violated if an attacker is also able to interpret the data gathered through snooping. Detection of passive attacks is very difficult since the operation of the network itself does not get affected. One way of preventing such problems is to use powerful encryption mechanisms to encrypt the data being transmitted, thereby making it impossible for eavesdroppers to obtain any useful information from the data overheard. Attacks of such type include Eavesdropping and Traffic Analysis & Monitoring. During eavesdropping, the information the attacker may target include the location, public key, private key or even passwords of the nodes. By traffic analysis and monitoring, packet transmission is analyzed to infer important information such as a source, destination, and source destination pair. The information collected could be used to initiate other active attacks.

2. Active Attacks

An active attack attempts to alter or destroy the data being exchanged in the network, thereby disrupting the normal functioning of the network. It can be classified into two categories external attacks and internal attacks. External attacks are carried out by nodes that do not belong to the network. These attacks can be prevented by using standard security mechanisms such as encryption techniques and firewalls. Internal attacks are carried out by compromised nodes that are actually part of the network. Since the attackers are already part of the network as authorized nodes, internal attacks are more severe and difficult to detect when compared to external attacks. There are many attacks in this category. Some of the active attacks surveyed in [22], [23] are discussed as follows:

Blackhole attack: In a black hole attack, a malicious node sends fake routing information, claiming that it has an optimum route and causes other good nodes to route data packets through the malicious one. For example, in AODV, the attacker can send a fake RREP (including a fake destination sequence number that is fabricated to be equal or higher than the one contained in the RREQ) to the source node, claiming that it has a sufficiently fresh route to the destination node. This causes the source node to select the route that passes through the attacker. Therefore, all traffic will be routed through the attacker, and therefore, the attacker can misuse or discard the traffic [23].

Grayhole attack: The grayhole attack has two phases. In the first phase, a malicious node exploits the AODV protocol to advertise itself as having a valid route to a destination node. In the second phase, the node drops the intercepted packets with a certain probability. This attack is more difficult to detect than the black hole attack. A gray hole may exhibit its malicious behavior in different ways. It may drop packets coming from (or destined to) certain specific node(s) in the network while forwarding all the packets for other nodes. Another type of grayhole node may behave maliciously for some time duration by dropping packets but may switch to normal behavior later [24].

Wormhole Attack: In wormhole attack, a malicious node receives packets at one location in the network and tunnels them to another location in the network, where these packets are resent into the network. This tunnel between two colluding attackers is referred to as a wormhole. It could be established through wired link between two colluding attackers or through a single long-range [29] wireless link. In this form of attack the attacker may create a wormhole even for packets not addressed to itself because of broadcast nature of the radio channel.

Flooding attack: In flooding attack, attacker exhausts the network resources, such as bandwidth and a node's resources, like computational and battery power or to disrupt the routing operation to cause severe degradation in network performance. For example, in AODV protocol, a malicious node can send a large number of RREQs in a short period to a destination node that does not exist in the network. Because no one will reply to the RREQs, these RREQs will flood the whole network. As a result, all of the node battery power, as well as network bandwidth will be consumed and could lead to denial-of-service attack [24].

2.5.3 Flooding attacks in AODV

Flooding-based attacks are one of the most dangerous attacks that aim to consume all network resources and thus paralyze the functionality of the whole network. Therefore, the objective of this paper is to investigate the threshold value of nodes and plus RREQ RATE LIMIT that happen during the nodes communications from the same source to detect flooding-based attacks in MANETs.

2.5.4 Types of flooding attacks

RREQ Flooding attack: In this form, the attacker node keeps flooding the network with requests (RREQs) for random nodes' IDs that do not exist or may exist in the network. Normal nodes keep forwarding these RREQs hoping to find a path of fake nodes. This is my focusing area for my thesis work.

Data flooding attack: Also called Sleep Deprivation Attack. In this form, two attacker nodes in the network start to transmit an enormous amount of fake data to each other in a high sending rate in order to consume the energy of each normal node that is a part of the path between the two attacker nodes.

SYN Flooding attack: In this form, the attacker node consumes normal nodes memory by continuously sending an enormous amount of synchronization packets to the victim node.

In AODV protocol, a malicious node can send a large number of RREQs in a short period to a destination node that does not exist in the network. Because no one will reply to the RREQs, these RREQs will flood the whole network. As a result, all of the node battery power, as well as network bandwidth will be consumed and could lead to denial-of-service attack. This is the case to select this area for my research work to minimize its problems occurs during nodes to nodes communications [24].

2.6 Impact of RREQ flooding attacks

- Drops more data packets.
- Consume node’s battery power.
- Wastage of nodes’ processing time, thus increasing the delay.
- Low throughput.
- Consume bandwidth.
- Making the destination node unreachable with the purpose of having no successful communication between source and destination.

When we say RREQ flooding attack is Consume bandwidth and battery power, the attackers broadcasts mass Route Request packets or sends a lot of attacking data packets to exhaust the communication bandwidth and node resource so that the valid communication cannot be kept. This leads consumes battery power, storage space and bandwidth.

Throughput refers to the number of bits transferred from one mobile node to another. As ad hoc network contains a number of mobile nodes, the data load increases, which causes a low throughput among the mobile nodes.

CHAPTER THREE: RELATED WORKS

In this section, a lot of authors works on which are related to this thesis work, that is, preventing route request flooding attacks, are examined. The area of focus and the limitations of these works are also discussed.

Opinder Singh et al. [25] developed a new model called SAODV to detect and isolate the RREQ Flooding attack in MANET. SAODV uses a statistical threshold to detect the attacker node, which depends on two parameters: the mean number of RREQs (MRREQs) made by different nodes in the network and the mean deviation from the mean of all RREQs (MDRREQs). After computing these two parameters, the value of the threshold is set. Any node that sends a number of RREQs higher than the threshold is considered as an attacker node and an alarm will be broadcasted to isolate this node. The results of SAODV showed a high Throughput that is near to the native AODV and a low delay that is also near to the native AODV.

Sheetal Jatthap et al. [26] proposed a technique to detect and isolate RREQ Flooding attacker nodes based on their energy. The proposed technique analyzes a node's energy consumption in the network without an attack and then analyzes a node's energy consumption after an attack. The analysis process is performed to determine max and min energy threshold. If the node's energy is equal to or less than the min energy threshold, then the node is dead. And if the sender node has a higher energy than the max threshold, it is considered as an attacker node and is then added to the blacklist in order to isolate it and to avoid communication with it. The results showed a lower protocol power consumption and a lower node power consumption compared to the native AODV under attack.

D. Srinivasa Rao et al. [27] proposed a technique to avoid the RREQ Flooding attack in MANET. The proposed technique depends on dividing the network into clusters to avoid any RREQ Flooding because only cluster head nodes are allowed to broadcast RREQs in the network. Any RREQ that comes from a normal node is dropped. The proposed technique is divided into three phases: Join Network, Cluster Head Election, and Path Cutoff. When a node joins a network in the Join Network phase, it identifies itself and joins the nearest cluster, and then it gets a Unique Identifier (UID). In the second phase, nodes are elected to be a cluster head to control communication between nodes. And in the third phase, when a node receives an RREQ not from a cluster head, the request is then dropped. The results showed a high Packet Delivery Ratio (PDR) that is almost the same as the native AODV but it also showed a higher overhead than the native

AODV.

Vrince Vimal et al. [28] developed a technique used to detect and prevent RREQ Flooding attack in MANET. The developed technique has a Detection and Prevention mechanisms. In Detection mechanism, the number of neighbor nodes is used to determine the value of the threshold, which is used to detect the malicious node. Any node that sends a number of RREQs more than the threshold is considered as a malicious node and is added to the blacklist to avoid communicating with it. In Prevention mechanism, neighbor nodes are notified about the malicious node by an alarm packet. To continue the communication normally, routes are modified by replacing any malicious node that forwards packets to destination nodes, with the nearest normal node. The results showed an increase in Packet Delivery Ratio (PDR) compared to native AODV under attack and a high Detection Rate of the malicious nodes.

Surendra Kumar et al. [29] developed an algorithm to prevent RREQ attack in MANET. Each node has three lists: whitelist, gray list, and blacklist. Whenever a node receives a request, it searches the sender in these three lists. If the sender is from the blacklist, the request is dropped, and if the packet is from a gray list, then it is checked if there is a black alarm broadcasted about the sender node. If such an alarm exists, the request is dropped; otherwise, the request is served. Finally, if the sender is from the whitelist, then the request is served. The judgment on nodes depends on the request number received from the node. If it is higher than the major threshold, then it is in the blacklist and a black alarm is broadcasted. If it is higher than the minor threshold, then it is in the gray list and a gray alarm is broadcasted.

Shashi Gurung et al. [30] proposed a novel approach to mitigate RREQ Flooding attack in MANET. The proposed approach is called F-IDS. It is divided into three phases: dynamic threshold calculation, confirmation, and resetting phase. In F-IDS, nodes are in the promiscuous mode to observe the nodes ’behavior in the network. In the first phase, after a period of time, each node calculates the threshold value based on the standard deviation of the received requests number. In the second phase, if nodes detect am is behaving node that broadcasts fake requests greater than the threshold, an alarm is broadcasted to all normal nodes to block this node and add it to the blacklist. In the third phase, nodes reset blocked nodes in the blacklist after a period of time, and only if a node has been blocked for three times, then this node will be blocked forever.

The results showed a high average Throughput that is near to the native AODV but a higher Normalized Routing Load than the native AODV.

Mahmoud Abu Zant and Adwan Yasin [31] Avoiding and Isolating Flooding Attack by Enhancing AODV MANET Protocol. According to the ideas of these researchers, they used dynamic value limit rather than static depending on the number of neighbor nodes (node connectivity). This is done in three ways:

If the number of neighbor nodes is less than half of the limit, then set the limit to limit/2 (the limit is 10 according to RFC 3561 (Which means, for example if node “A” has less than 5 connectivity, set the limit =5 so, if this node make a request more than 5 per second, it can be an attacker and ignore it.) 2. If the number of neighbor nodes is greater than half of the limit and less than limit *1.5, then set the limit equal to number of connectivity. (Which means, for example if node “A” has between 5 and 15 connectivity, set the limit = equal to number of connectivity i.e., if this node make a request 6 per second, limit=6.) 3. If the number of neighbor nodes is greater than 15, then set the limit equal to 15. (Which means, for example if node “A” has 16 connectivity, set the limit = 15 i.e., if this node make a request more than 15 per second, it can be an attacker and ignore it.) The lack of this paper is the number of node connectity cannot determine whether the requesting node is an attacker node or not.

□Here is the general equation of this paper:

$$\mathbf{LV}= \begin{cases} \mathbf{NON,} & LV/2<NON<LV*1.5 \\ \mathbf{LV/2,} & NON<=LV/2 \\ \mathbf{LV*1.5,} & NON>=LV*1.5 \end{cases}$$

Note: **LV** is the limit value (the default value of Lv is 10).

NoN is the number of neighbor nodes.

The limitation in [25], [26] is that they depend on a static value as a threshold to detect the attacker node in the network, which should be a dynamic value. The limitation in [27], [28], [29] is that an alarm message is broadcasted to normal nodes after the detection of an attacker node in the network, which makes the network vulnerable to a blackmail attack because a blackmail attacker node can broadcast false alarm messages containing normal nodes IDs to isolate them from other normal nodes in the network. The limitation in [30] is that the detection of an attacker node depends on the exchange of information about other nodes, which makes the network vulnerable to false information exchange by cooperative attacker nodes. The limitation in is that the proposed model depends on clustering the network to detect the attacker node and it is known that clustering has a high overhead in MANET. That is why some network environments avoid clustering.

CHAPTER FOUR: DESIGN OF THE PROPOSED SOLUTION

The proposed prevention mechanism is discussed in detail in this section. Before that, the network Simulator and the implementation of AODV in NS-2 are introduced as follows.

4.1 Network Simulator 2 (NS2): Features & Basic Architecture of NS2

What is Network Simulator (NS2)?

Network Simulator (Version 2), widely known as NS2, is simply an event-driven simulation tool that has proved useful in studying the dynamic nature of communication networks. Simulation of wired as well as wireless network functions and protocols (e.g., routing algorithms, TCP, UDP) can be done using NS2. In general, NS2 provides users with a way of specifying such network protocols and simulating their corresponding behaviors [30].

4.1.1 Features of NS2

1. It is a discrete event simulator for networking research.
2. It provides substantial support to simulate bunch of protocols like TCP, UDP, https, AODV and DSR.
3. It simulates wired and wireless network.
4. It is primarily UNIX based.
5. Uses TCL as its scripting language.
6. Otcl: Object oriented support
7. Tclcl: C++ and otcl linkage
8. Discrete event scheduler

4.1.2 Basic Architecture of NS2

NS2 consists of two key languages: C++ and Object-oriented Tool Command Language (OTcl). While the C++ defines the internal mechanism (i.e., a backend) of the simulation objects, the

OTcl sets up simulation by assembling and configuring the objects as well as scheduling discrete events.

The C++ and the OTcl are linked together using TclCL.

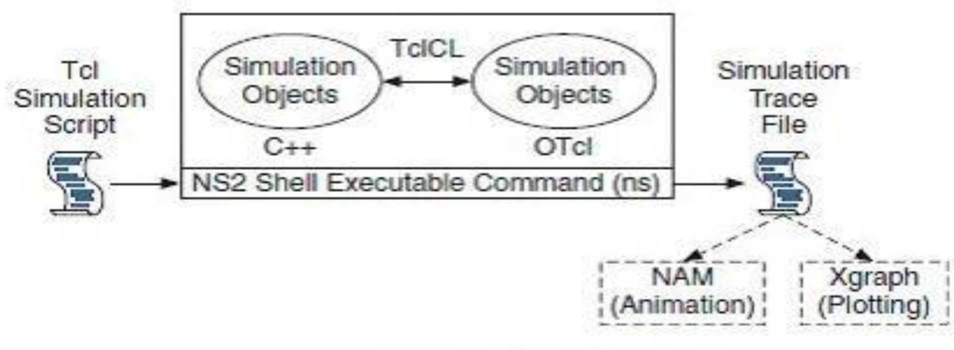


Figure 4. 1 Basic architecture of NS2 [29], [30].

4.2 Why NS2 is preferable for our implementation?

There are many network simulators which has their own features to distinguish it from other. We should focus on choosing the best one that gives the best results. NS2 is the most common simulator tool, in this thesis we used NS2 to design the network, which made communication within the network with different routing protocols [32].

4.3 Overview of AODV Implementation in Ns2

NS2 implements, as described in [33], a routing protocol using routing agents which are used to create, transmit, receive, and destroy routing packets. NS2 declares a C++ class AODV and class AODV derives from class Agent and inherits three important attributes and behaviors: (1): a pointer “target_” which points to a link layer object, (2) a function allocpkt() which can be used to create packets and (3) a packet reception function recv(p,h). Among these attributes and behaviors, only packet reception is overridden by class AODV.

Main C++ files for AODV are stored in the directory of ~ns/aodv/ and include the following.

- ✓ Aodv.cc and aodv.h: These files include the main definitions of AODV routing agents.

- ✓ Aodv_packet.h: this file defines the packet header of AODV protocol including RREQ, RREP and RERR messages.
- ✓ Aodv_rtable.cc and aodv_rtbale.h: these files handle the routing entries and tables of the AODV routing protocol.
- ✓ Aodv_rqueue.cc and aodv_rqueue.h: define the buffer which stores data packets during a routing discovery procedure.

Other AODV related classes include Agent, Timers and routing information. The Agent class is responsible for creating, sending, receiving, processing, and destroying routing packets and AODV uses class AODV for this purpose. The timers take care of the time-driven actions and include classes like Broadcast Timer, Hello Timer, NeighbourTimer and RoutecacheTimer. Routing information is handled using two classes: aodv_rt_entry and aodv_rt_table classes. For the purpose of route discovery, AODV uses SendRequest (nsaddr_t dst), SendReply (nsaddr tipdst…), recvRequest (Packet *p) and recvReply (Packet *p) functions to send RREQ, send RREP, receive RREQ and receive RREP respectively. In the case of the send functions, the RREQ or RREP is first build and the respective fields are filled with the right values before they are sent to neighbors.

The following box shows the default values for some important parameters associated with AODV protocol operations and what we are used in our thesis [31].

NET_TRAVERSAL_TIME= 2 * NODE_TRAVERSAL_TIME * NET_DIAMETER
NODE_TRAVERSAL_TIME = 40 milliseconds

NET_DIAMETER =35 **(**the maximum possible number of hops between two nodes)

NET_DIAMETER measures the maximum possible number of hops between two nodes in the network. NODE_TRAVERSAL_TIME is a conservative estimate of the average one hop traversal time for packets and should include queuing delays, interrupt processing times and transfer times.

The following is the equation designed for calculating threshold value to design the new prevention algorithm in AODV protocol.

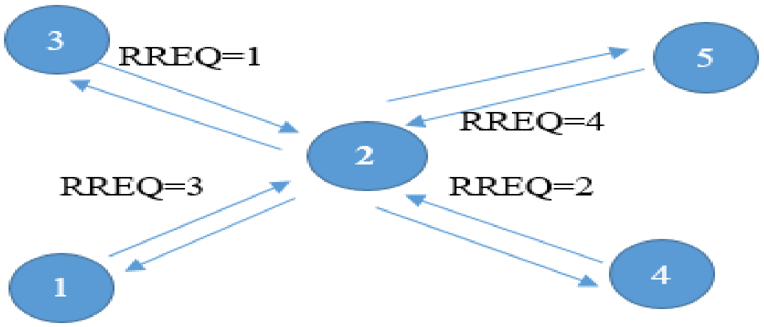


Figure 4. 2 Number of RREQ

As an example the figure above shows that:

Number of RREQ of node 1, 3, 4 and 5 with in NET TRIVERSALTIME to node 2 are 3, 1, 2 and 4 respectively. So, the threshold value is calculated by node 2 is: (3+1+2+4)/5=2 then, add RREQ rate limit of default AODV which is 2+10=12, finally the threshold value is= 12. So, any nodes that makes rreq greater than 12 within per second is counted as an attacker node.

Note that, the main reason of using the above equation is the one which increases the performance metrics in AODV such as packet delivery ratio, residual energy, and throughput and decreases the end to end delay when we have compared to the previous work such as AIF-AODV and pure AODV. Which means that, when we use threshold values starting from half of RREQ rate limit of default AODV (RREQ RATE LIMIT/2) to 10 respectively, the performance of AODV is still not much good. Especially, packet delivery ratio, residual energy, throughput and end to end delay are not good when compare the result with the several researcher works. Due to this we have use the above equation to detect rreq flooding attack which gives the best result and success our proposed work.

The implementation of AODV routing protocol in NS-2 is also briefly described in [35][36].

Threshold =$number\ of\ rreq\ of\ \frac{n0+n1+n2+\cdots nn}{n}$ +*RREQ rate limit*

Which means, threshold= mean of rreq+ RREQ *rate limit*

NOTE:

n*: implies total number of nodes.*

RREQ rate limit =10 (default value of normal AODV).

NET TRAVERSAL TIME calculated as:

2 * NODE_TRAVERSAL_TIME * NET_DIAMETER

4.4 Design and Implementation of Mitigation Mechanism

4.4.1 Design of the Mitigation Mechanism

The RREQ flooding attack occurs at the route discovery stage. So in this case the existing protocol is modified to include the proposed scheme.

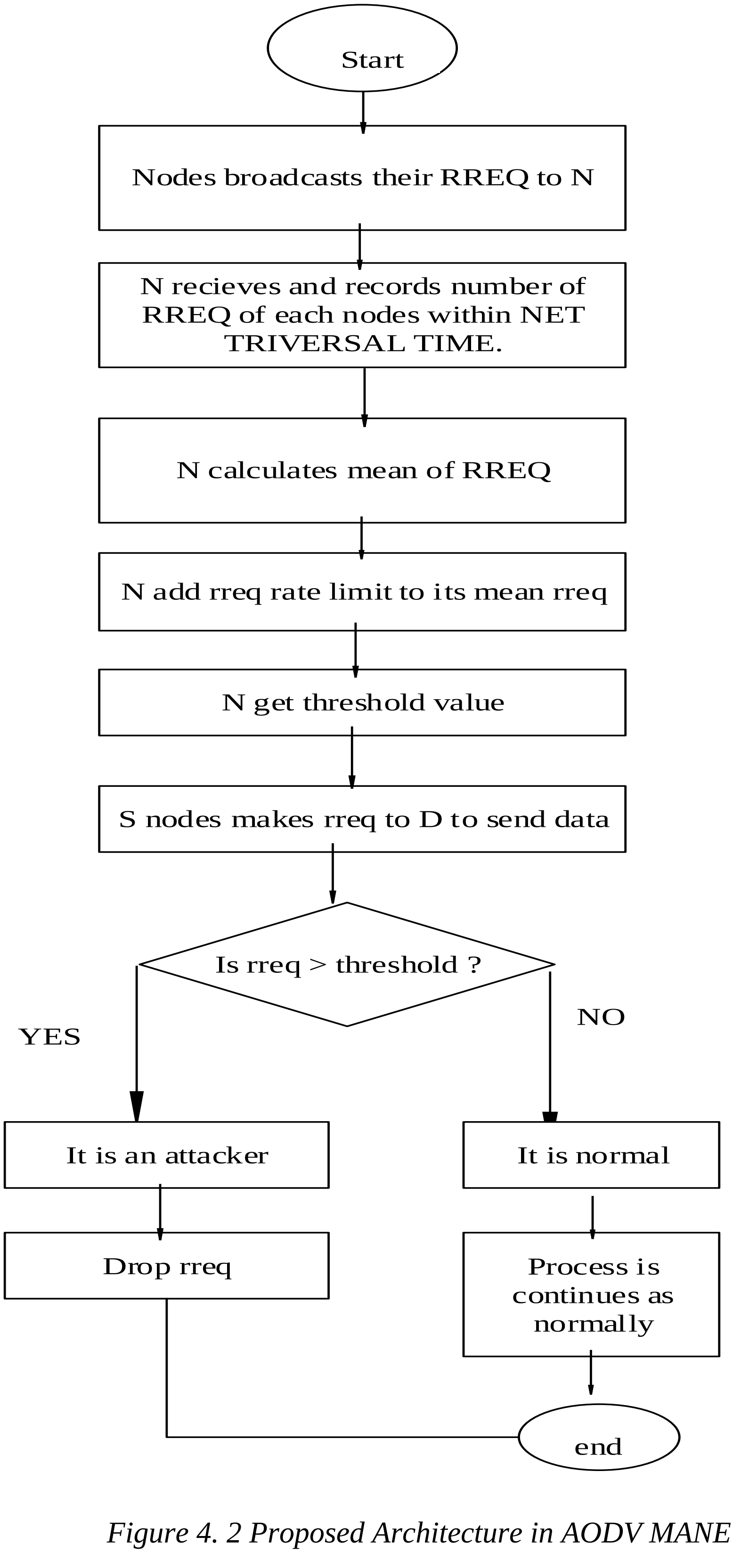

Start
Nodes broadcasts their RREQ to N
N recieves and records number of RREQ of each nodes within NET TRIVERSAL TIME.
N calculates mean of RREQ
N add rreq rate limit to its mean rreq
N get threshold value
S nodes makes rreq to D to send data
Is rreq > threshold ?
YES
NO
It is an attacker
It is normal
Drop rreq
Process is continues as normally
end

Figure 4. 2 Proposed Architecture in AODV MANET

4.4.2 Proposed algorithm for mitigation of flooding attack in AODV

```
Input: Node N, S, rreq_rate_limit

Process:

N nodes sends number of rreq with in NET TRAVERSAL TIME to nodes.
N node(s) records number of rreq of n nodes to get threshold.
N node calculates threshold based on number of rreq generated from each nodes in
    $(n0+n1+n2+n3+…nn)/N + rreq\_rate\_limit$
S make flooding RREQ to neighboring nodes.
IF (S node rreq <= threshold)
    Declare itself as normal_node
ELSEIF (S node rreq > threshold)
    Declare itself as attacker_node
    Drops_requests
ENDIF

Output: normal_node, attacker_node, Drops_requests, threshold
```

4.5 Implementation

To implement this mitigation, the algorithms of the flooding attack and the prevention mechanism are incorporated to the existing AODV routing protocol by modifying the C++ code of the protocol. In implementing the flooding attack, aodv.cc and aodv.h files are modified to include the malicious behavior to the protocol. Class Flood Timer is used to implement a timer that will be used to schedule the flooding attack. Basically, the SendRequest and recvRequest functions of AODV, which are found in aodv.cc file, are modified to implement the mitigation mechanism.

4.6 AWK

To process and extract important information from huge amount of data or text, a scripting language is necessary and essential. From those scripting languages for our research we prefer to use AWK programming language because it is suitable for processing huge amount of network trace files to produce performance results.

The trace file is meaningless unless the analysis is done in such a way that meaningful result is obtained from the output file. For each QoS metrics we wrote AWK script that obtains the performance information from the trace file. The analyzed results obtained for each QoS metrics are recorded using **gnuplot** and finally plotted as a graph. We wrote AWK script to find QoS metrics values by obtaining network communication information from the trace file.

CHAPTER FIVE: SIMULATION, RESULTS AND RESULT ANALYSIS

In this Chapter, we present the performance evaluation of Proposed AODV algorithm. First simulation scenarios and model will be discussed; the network topology and movement of nodes with traffic models will be discussed next. Then detail simulation results and analysis will be presented.

5.1 Simulation Model

In the following sections we are present the models (network, mobility and traffic models) and the parameters used in the simulations.

5.1.1 Network Model

The physical network of a MANETs consists of mobile nodes such as laptops, PDAs and wireless phones. It is self-configuring and there is no need for other infrastructure. The communicating devices have routing capabilities and operate both as hosts and routers to forward data packets to each other. They move freely in a random way and usually multiple hops are needed to exchange data between each two nodes. To model the network we used a rectangular and constant simulated area of 1200x1200 meters. We also used typical NS2 parameters like the standard Two-rayGround as a radio propagation model for the Wireless channels and Omni-Directional Antenna model. The network interface type is the standard IEEE 802.11. To observe the effect of increasing the number of communicating devices we use a varying number of mobile nodes (20, 40 and 60).

5.1.2 Mobility Model

The mobility model describes how speed, acceleration and direction of the node changes over time. It is very important as it changes the characteristic of the mobile nodes and thus effects network and routing protocol performance. In order to check the performance of a protocol for an ad hoc network, the protocol should be tested under realistic conditions such as limited transmission range, limited buffer space for storage of messages and realistic movement characteristics of mobile nodes. There are various mobility models [4] such as Random Walk Mobility Model, Random Waypoint Mobility Model, Two ray ground mobility Model etc. Two Ray Ground model (algorithm) which is the one used for these simulations.

- ✓ **Mobility Management**

AODV routing protocols is preferred due to less control overhead and scalability, but it suffer from frequent link breakages due to the high-mobility of the nodes. To reduce the link breakages and get a stable route, a new reactive routing protocol is proposed that is two ray ground mobility Model-aware. The proposed Mobility and Direction Aware Ad-hoc On Demand Distance Vector routing protocol aims to handle the mobility and direction factors in ad-hoc networks. Mobility and Direction Aware Ad-hoc On Demand Distance Vector guides the route discovery and route reply depending on the speed of the participating nodes and their directions.

Mobility and Direction Aware Ad-hoc On Demand Distance Vector prevents and manages high mobility nodes (their speeds are more than a specific threshold) from participating in the route discovery process. In the route reply process, once the route request packet reaches the destination or any intermediate has an active route toward the destination. Then the new scheme will apply an algorithm to select the best path form different paths, through which requests were received, depending on the speed and the direction for the participating nodes in the routing process

5.1.3 Traffic Model

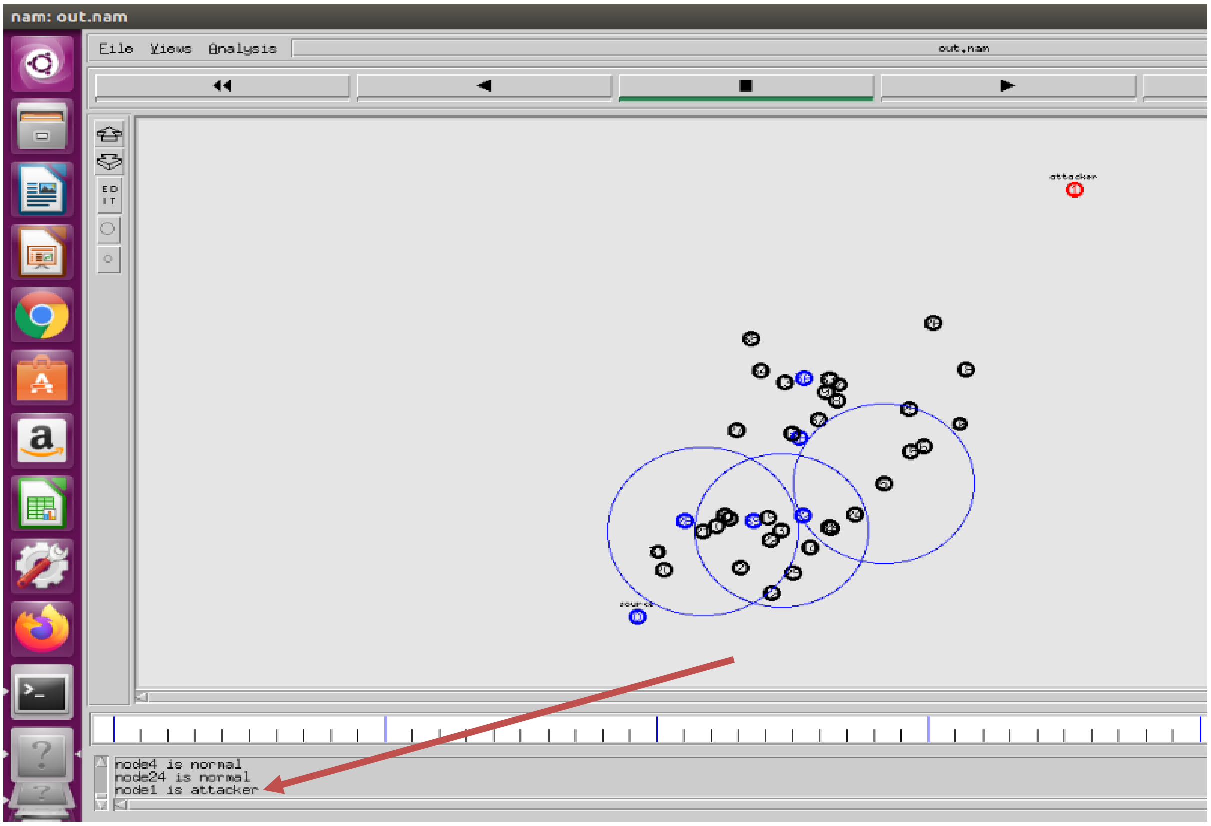
Traffic density is another key parameter that affects the overall network and protocol behavior in MANETs. In other words, number of connections between the mobile nodes and other parameters like packet size, packet rate etc. influence the performance metrics we are interested in. we have used CBR (UDP) traffic as it doesn't vary in the different simulations. Using CBR for comparison purposes is important in order to get fair results. Varying traffic (i.e. TCP) could make the load unpredictable and corrupt the simulation results. We have used 512 Bytes constant packet size. To have a good modeling of the traffic flows the source-destination pairs must be chose and spread randomly over the network.

Examined protocol	AODV,AIF-AODV and MRREQF-AODV
Total number of nodes	20,40 and 60
Simulation time	10s
Simulation area	1200*1200
Connection	CBR
Antenna type	Antenna/OmniAntenna
Flooding attack interval	0.06 in per second
Mobility model	Two ray ground model
Data packet size	512

Table 5. 1 Simulation parameters value.

5.2 Simulation Setup

The goal of all the simulations in this paper is to present a quantitative relation between network performance metrics such as packet delivery ratio, throughput and e2e delay and other varying network parameters like the number of nodes, number of connections, mobility speed and pause time. Traffic and mobility files are included in the simulation at the time of execution [32]. Every simulation is run for 10 seconds. To run the simulations we have used NS-2 version 2.35 built on Ubuntu 16.04 Linux. Figure 5.1 shows a simulation screenshot taken from NAM. We used AWK scripts to process the trace files and plot the graphs of the performance metrics.



```
node4 is normal
node24 is normal
node1 is attacker
```

Figure 5. 1 simulation setup of nam file

5.2.1 Performance Metrics

There are many network performance metrics which can be evaluated in order to get an overview of the performance of routing protocols [33]. In this paper, AODV with malicious and without malicious nodes, performance is analyzed and compared using the metrics described below End to End Delay.

End to end delay on network refers to the time taken for a packet to be transmitted across a network from source to destination device. According to the obtained performance of the networks the proposed algorithm consumes less time for propagation of data as compared to the traditional routing protocol under attack conditions.

- ✓ **Throughput**

It is defined as packets delivered over the total simulation time. It is the usual rate of successful delivery of a message over a communication medium. The throughput is measured in the form of bits or bps, and occasionally is measured in the form of data packets per time slot or data packets per second.

It is calculated as follows:

Throughput = (No. delivered packet * packet size)/total duration of simulation.

Higher Throughput means better performance of the protocol

- ✓ **Packet Delivery Ratio (PDR)**

Packet delivery ratio (PDR) provides information about the performance of any routing protocols, where PDR is estimated using the formula given [34].

$$\text{Packet delivery ratio} = \frac{\text{total delivered packets}}{\text{Total sent packets}} * 100$$

According to the obtained results the performance of the proposed algorithm is much better in attack conditions.

- ✓ **Average end-to-end Delay (AEED)**

End-to-end (EED) delay is the average delay between sending the data packet by the source and its receipt at the corresponding receiver for a single scenario. This includes all the delays caused during route acquisition, buffering and processing at intermediate nodes, and retransmission delays at the MAC layer for a given simulation time. This can be expressed as:

AEED = Time packet correctively received – time packet sent/Total number of packet received.

- ✓ **Average Residual Energy (ARE)**

Average residual energy measures the average of remaining energy in every node in the network.

It can be computed as follows:

ARE = $\boldsymbol{RE/N}$

Where;

*RE is the residual energy and **N** is the number of nodes in the network*

5.3 Simulation results and discussion

In this section the simulation results are shown with respect to the metrics discussed in the previous section. The section also discusses the results based on the table and graph. The following graphs were plotted using the average values of PDR, Average E2E delay, average residual energy and Throughput under varying number of nodes.

Table 5.2 shows the numeric results of comparison between AIFAODV and MRREQF_AODV in terms of PDF, Throughput, End to EndDelay and ARE.

Number of nodes	Normal AODV	AIF-AODV(previous work)	MRREQF-AODV (Proposed Work)
	Packet Delivery Ratio (PDR) (%)		
20	16.46	17.9	93
40	16.35	19.2	93
60	24.41	11.1	86
	Throughput(kbps)		
20	86.75	173.4	261.32
40	39.12	195.4	266.1
60	107.61	113.1	246.93
	Avg of End to EndDelay(ms)		
20	1.3726	1.124	0.0165897
40	1.361	1.034	0.0184648
60	1.28295	1.180	0.0098557
	Avg Residual Energy(ARE)(joule)		
20	2.67	1.680	2
40	0.6675	1.087	2.0125
60	0.296667	0.375	1.9175

Table 5. 2 Simulation results of the flooding attack between AIF-AODV vs MRREQF-AODV.

As shown in table 5.2 above, the result of PDR in AIF- AODV is the lower when compared to MRREQF-AODV, especially when the number nodes increased. It is clear that the effect of the attack increases when the number of nodes increases because of the rebroadcasting of fake requests and the overhead of finding the fake nodes in the network. The result of PDR in native AODV after mitigating is the highest when there is no Flooding attack in the network.

As shown in table 5.2 above, the result of Throughput in native AODV with RREQ flooding attack is decreasing, while the number of nodes increases as a result of the rebroadcasting of fake requests. The Flooding attack will lead to congestion in the network which also leads to dropping and delaying normal packets, which in turn will affect the Throughput and PDR. The result of MRREQF-AODV shows a higher Throughput than native AODV under Flooding attack and AIFAODV.

As shown in table 5.2, the result of End to End Delay in native AODV with RREQ flooding attack is increasing, while the number of nodes increases because of the congestion generated by the flooding node. Normal packets will get dropped or delayed, which will increase the End to End Delay. The result of AIF AODV shows a higher End to End Delay than native AODV after detects Flooding attack because MRREQF- AODV detects and isolates the attack node in the network.

As shown in table 5.2, the result of ARE in native AODV when there is a Flooding attack is the lowest, especially when the number of nodes increases as the Flooding attack consumes the energy of nodes by keeping them busy in rebroadcasting fake requests in the network. The result of ARE in native AODV when there is no Flooding attack in the network is the highest. The result of MRREQF-AODV shows a higher ARE than native AODV under Flooding attack because MRREQF-AODV prevents the Flooding attack in the network.

5.3.1 Simulation results

The following graphs were plotted using the average values of PDR, Average E2E delay, average residual energy and Throughput under varying number of nodes.

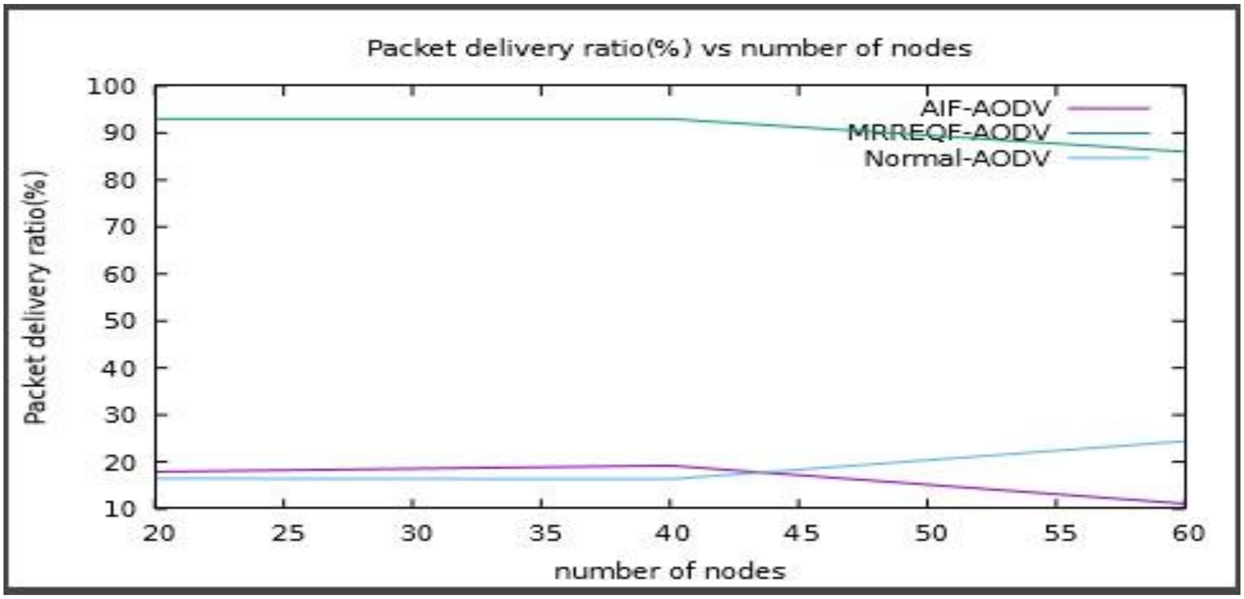


Figure 5. 2 PDR vs number of nodes.

As shown in Figure 5.2 above, the result of PDR in AODV and AIF- AODV is the lower when compared to MRREQF-AODV, especially when the number nodes increased.

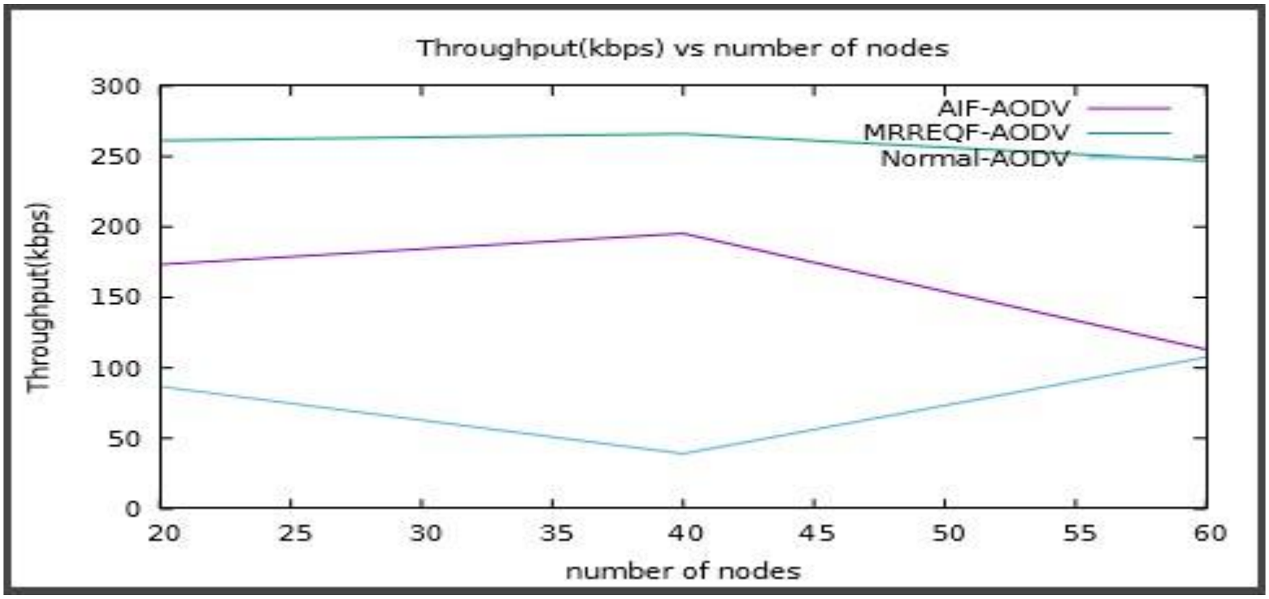


Figure 5. 3 Throughput vs number of nodes.

As shown in Figure 5.3 above, the result of throughput in AODV and AIF- AODV is the lower when compared to MRREQF-AODV, especially when the number nodes increased.

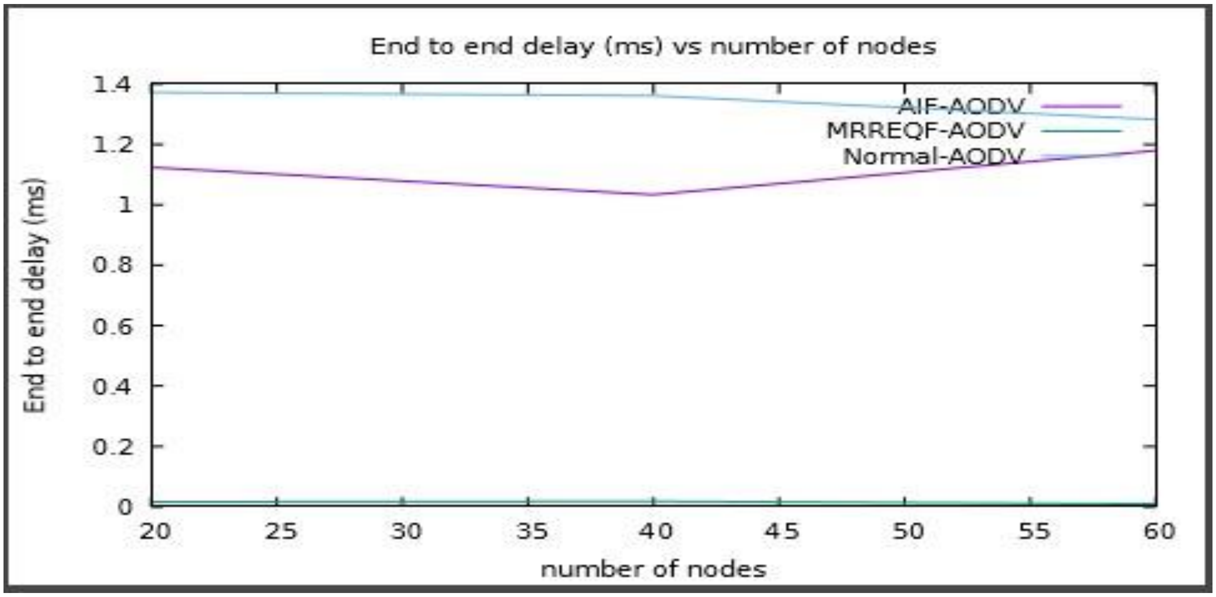


Figure 5. 4 End to End Delay vs number of nodes.

As shown in Figure 5.4 above, the result of end to end delay in Normal AODV and AIF- AODV are the higher when compared to MRREQF-AODV, especially when the number nodes increased.

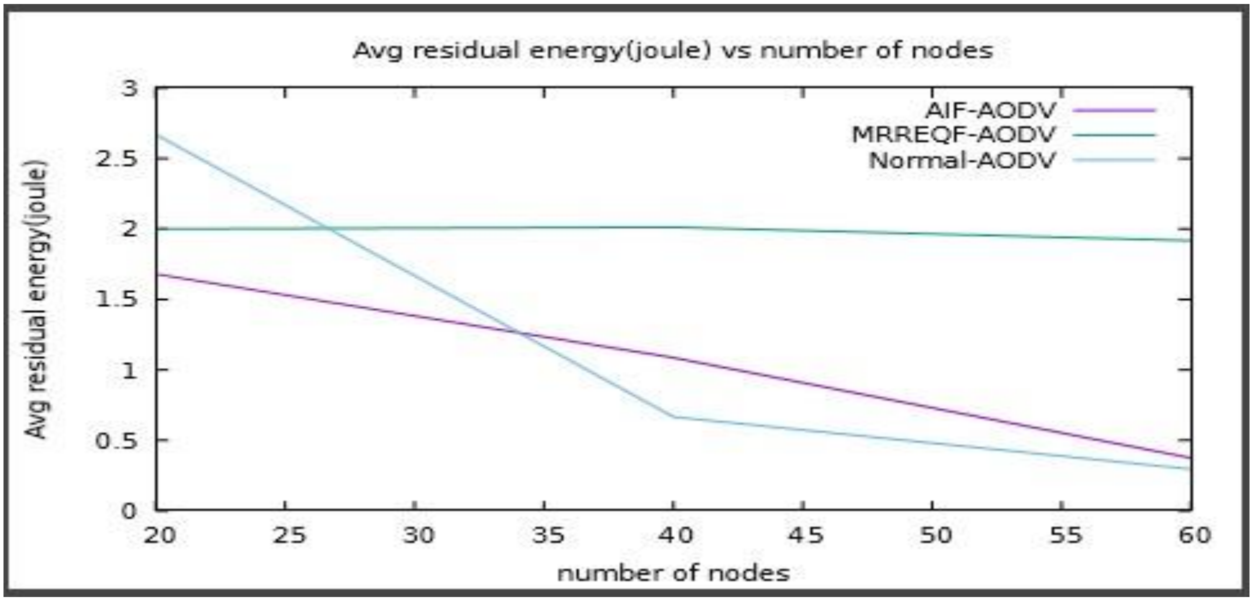


Figure 5. 5 ARE vs number of nodes.

As shown in Figure 5.5 above, the result of MRREQF-AODV shows a higher ARE than AIFAODV and AODV under Flooding attack because MRREQF-AODV prevents the Flooding attack in the network.

Analyzing overall Simulation results by graph.

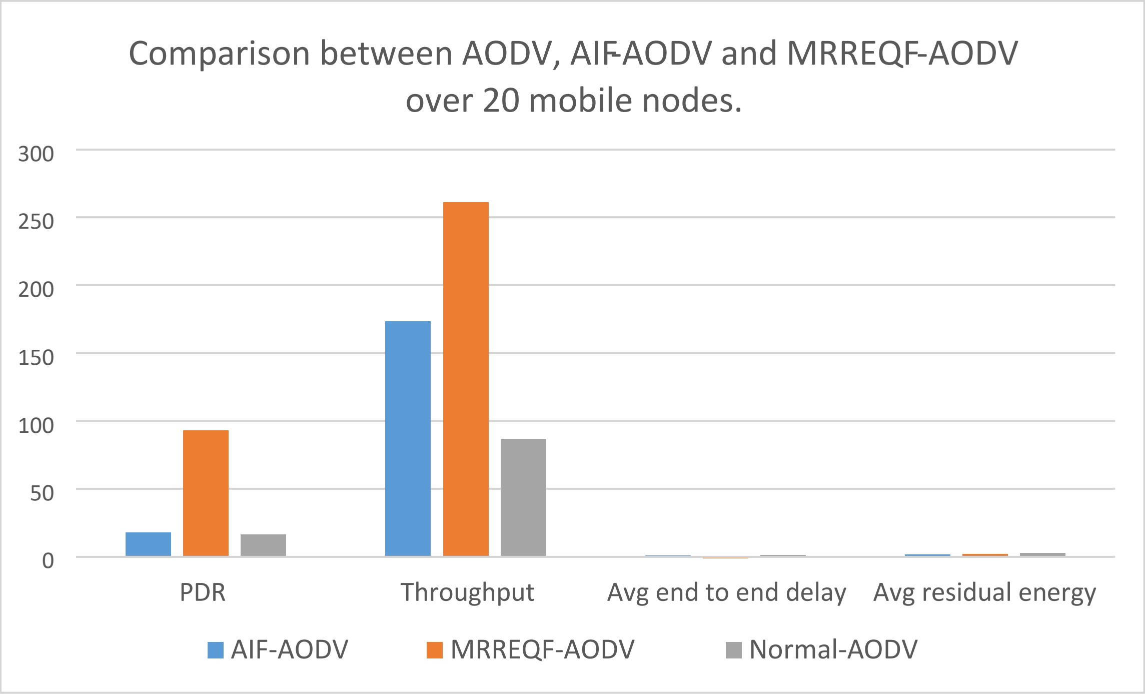


Figure 5. 6 Comparison between AODV, AIF-AODV and MRREQF-AODV over 20 mobile nodes.

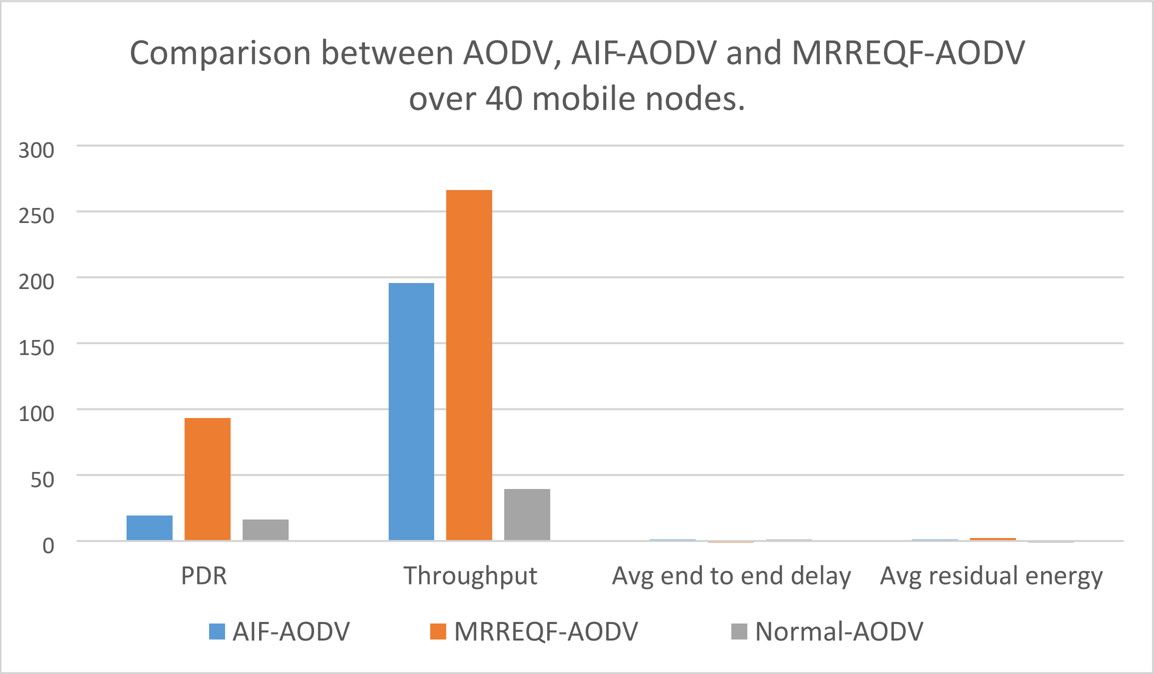


Figure 5. 7 Comparison between AODV, AIF-AODV and MRREQF-AODV over 40 mobile nodes.

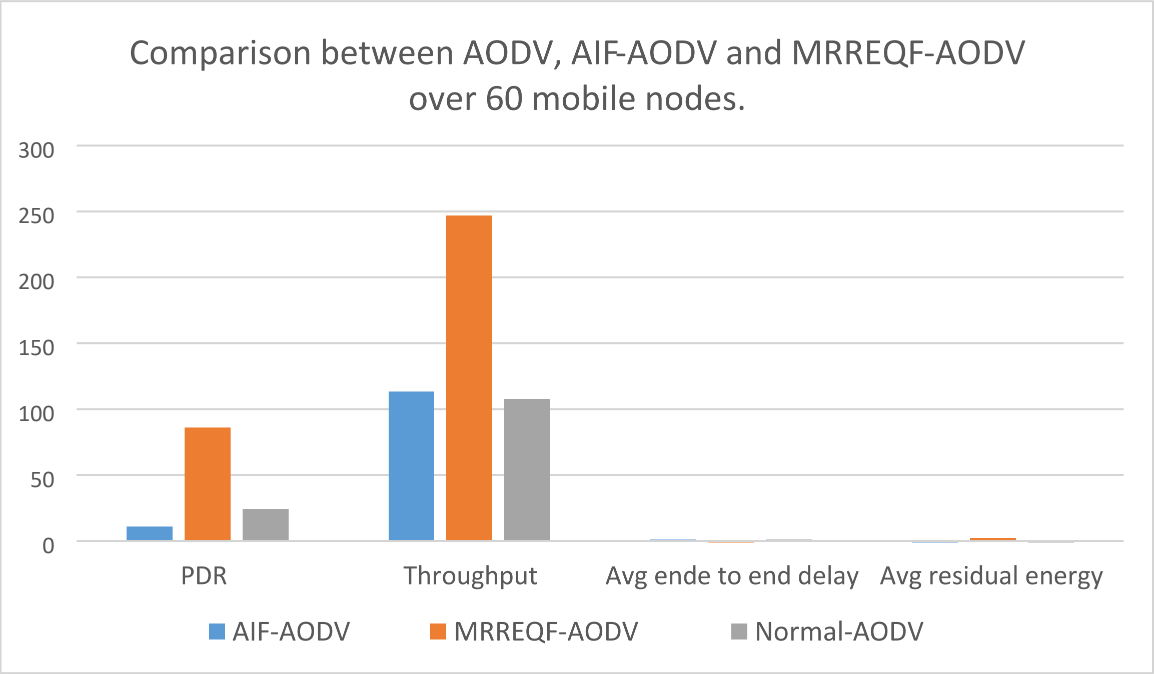


Figure 5. 8 Comparison between AODV, AIF-AODV and MRREQF-AODV over 60 mobile nodes.

In the given figure 5.6-5.8, the graph shows that the MRREQF-AODV packet delivery ratio and throughput is greater in 20, 40 and 60 mobiles nodes than normal AODV and AIF-AODV. Also Avg end to end delay in MRREQF-AODV within 20, 40 and 60 mobile nodes are less when compare to normal AODV and AIF-AODV which meets my objective and average residual energy in MRREQF-AODV is greater than AIF-AODV.

- ✓ **Performance Analysis of proposed work, AIF-AODV and Normal AODV**

As shown in figures above, the performance of the three ad hoc routing protocols AODV, MRREQ and AIF-AODV have been compared with twenty, forty and sixty maximum number of nodes. One can see that, the proposed algorithm that we are designed have good performance on behalf of prevention and detecting route request flooding attack in AODV routing protocol. Because, this prevention mechanism detected those route request flooding attack generated by each nodes are checked by each nodes on each paths before broadcasting the requests to its neighboring nodes. So that, the proposed algorithm have an ability to isolate this attacker from the network. This method also have a chance to detect an attacker from the communication if the requesting node is a new requester[33], [34].

- ✓ **Performance evaluation as number of node increased**

As a general, when the number of nodes increase in proposed algorithm, packet delivery ratio and throughput are almost the same. But in case of end to end delay and residual energy, the performance is decreased as the number of nodes increased. Because as the number of nodes increased in the network, the overhead of the network is increase which makes the communication between the nodes busy[35].

CHAPTER SIX: CONCLUSIONS, CONTRIBUTION, BENIFICIARY AND FUTURE WORK

6.1 Conclusions

The mobile ad hoc network is kind of wireless network and the network devices are connected through each other through this wireless links. The due to wireless connectivity among the network nodes the network is enabled to be support the mobility in network. Due to mobility and ad hoc configuration of network the network is always suffers from the performance and the security issues. In this research the proposed work is intended to provide a secure routing protocol and the high performance network [36]. Therefore the work involves the security investigation of the mobile ad hoc network and their performance enhancement.

Thus in order to investigate the security issues there are a number of research articles are explored and the attacks and their effect in network are studied. After that a crucial attack namely RREQ flooding attack is selected for finding the effective solution. Thus using the AODV routing protocol improvement a new secure routing protocol is developed. The proposed secure routing protocol first evaluates the threshold (variance) values are estimated then it compare with node's PDR, throughput, end to end delay and residual energy. These threshold values are used for taking decisions in network for discovering the secure path which protect from RREQ flooding attack in network.

The implementation of the proposed routing protocol is performed in network simulator 2 environment and using the generated trace files the performance outcomes are evaluated. During results evaluation the end to end delay, packet delivery ratio, throughput and residual energy is expected and compared with the previously work namely AIF-AODV available routing protocol.

The performance outcomes demonstrate the effectiveness of the proposed routing protocol as compared to other methods available. According to the given performance summary the proposed routing protocol is much adoptable as compared to the AIF-AODV routing protocols[37], [38].

6.2 Contribution

The following are the main contributions of the proposed RREQ flooding attack prevention mechanism.

- ✓ Attackers are prevented from imposing route request flooding attacks by completely isolating them; they can be identified as malicious by using number of rreq generated with in net traversal time from each nodes to Destination (mean rreq) and plus RREQ RATE LIMIT of default AODV.
- ✓ Increase the performance metrics of AODV especially the performance of PDR by isolating malicious node(s) from the network.

6.3 Beneficiary

This research will significantly contribute to the area of wireless communication in MANETs for effective communication because as long as the effect of attacker in communication are minimized and the QoS is enhanced finally the user will be satisfied with the service. This work will facilitate the services of Manet's applications and the most important application areas that will benefit from this work are emergency scenarios like military environments, multimedia, and in education[39], [40].

6.4 Future Work

Our proposed work does not cover the following tasks and it needs further study:

- ✓ Protecting of other types of attackers in AODV.

Reference

[1] K. S. Varsha and S. N. M. Raj, “Applications, Challenges and Protocols of MANETs: A Review,” *Asia-pacific J. Converg. Res. Interchang.*, vol. 4, no. 1, pp. 21–29, 2018, doi: 10.21742/apjcri.2018.03.03.

[2] S. Habib, S. Saleem, and K. M. Saqib, “Review on MANET routing protocols and challenges,” *Proceeding - 2013 IEEE Student Conf. Res. Dev. SCOReD 2013*, no. December, pp. 529–533, 2013, doi: 10.1109/SCOReD.2013.7002647.

[3] S. J. Lee, E. M. Belding-Royer, and C. E. Perkins, “Scalability study of the ad hoc on-demand distance vector routing protocol,” *Int. J. Netw. Manag.*, vol. 13, no. 2, pp. 97–114, 2003, doi: 10.1002/nem.463.

[4] K. K. Vashisth and K. C. Rojhe, “Distribution of Margin Among Intermediaries; Disintermediation and the Contemporary Entrepreneurial Flow,” vol. 2, no. 4, pp. 1–8, 2016, doi: 10.16962/EAPJMRM/issn.

[5] M. M. Alani, “MANET security: A survey,” *Proc. - 4th IEEE Int. Conf. Control Syst. Comput. Eng. ICCSCE 2014*, no. November, pp. 559–564, 2014, doi: 10.1109/ICCSCE.2014.7072781.

[6] M. A. Abdelshafy and P. J. B. King, “Analysis of security attacks on AODV routing,” *2013 8th Int. Conf. Internet Technol. Secur. Trans. ICITST 2013*, pp. 290–295, 2013, doi: 10.1109/ICITST.2013.6750209.

[7] O. Singh, J. Singh, and R. Singh, “SAODV : Statistical Ad hoc On-Demand Distance Vector Routing Protocol for Preventing Mobile Adhoc Network against Flooding Attack,” vol. 10, no. 8, pp. 2457–2470, 2017.

[8] T. Pandikumar and H. Desta, “RREQ Flooding Attack Mitigation in MANET Using Dynamic Profile Based Technique,” vol. 7, no. 6, pp. 12700–12705, 2017.

[9] N. Agrawal and U. Dwivedi, “Improved route relability to overcome route flooding attack in manet,” *Int. J. Appl. Eng. Res.*, vol. 12, no. 10, pp. 2497–2501, 2017.

[10] V. Vimal and M. J. Nigam, “Plummeting flood based distributed-DoS attack to upsurge networks performance in ad-hoc networks using neighborhood table technique,” *IEEE Reg. 10 Annu. Int. Conf. Proceedings/TENCON*, vol. 2017-Decem, pp. 139–144, 2017, doi: 10.1109/TENCON.2017.8227851.

[11] S. Kumar and S. Alaria, “Prevention in Sleep Deprivation Attack in,” vol. IV, no. Ii, pp. 139–144, 2015.

[12] S. Bhalodiya and K. Vaghela, “Enhanced Detection and Recovery from Flooding Attack in MANETs using AODV Routing Protocol,” *Int. J. Comput. Appl.*, vol. 125, no. 4, pp. 10–15, 2015, doi: 10.5120/ijca2015905878.

[13] M. Rmayti, Y. Begriche, R. Khatoun, L. Khoukhi, and D. Gaiti, “Flooding attacks detection in MANETs,” *2015 Int. Conf. Cyber Secur. Smart Cities, Ind. Control Syst. Commun. SSIC 2015 - Proc.*, 2015, doi: 10.1109/SSIC.2015.7245675.

[14] A. Panwar, D. S. Rao, and G. Sriram, “Combined Approach for Detection and Prevention of Flooding and Black-hole Attack in MANET,” *Int. J. Eng. Appl. Sci.*, vol. 4, no. 4, pp. 83–89, 2017.

[15] C. E. Perkins, M. Park, and E. M. Royer, “Ad-hoc On-Demand Distance Vector Routing,” *Proc. 2nd IEEE Work. Mob. Comput. Syst. Appl.*, pp. 90–100, 1997.

[16] M. Abu Zant and A. Yasin, “Avoiding and Isolating Flooding Attack by Enhancing AODV

MANET Protocol (AIF-AODV)," *Secur. Commun. Networks*, vol. 2019, 2019, doi: 10.1155/2019/8249108.

[17] B. Kannhavong, H. Nakayama, Y. Nemoto, N. Kato, and A. Jamalipour, "A survey of routing attacks in mobile ad hoc networks," *IEEE Wirel. Commun.*, vol. 14, no. 5, pp. 85–91, 2007, doi: 10.1109/MWC.2007.4396947.

[18] L. Abusalah, A. Khokhar, and M. Guizani, "A survey of secure mobile Ad hoc routing protocols," *IEEE Commun. Surv. Tutorials*, vol. 10, no. 4, pp. 78–93, 2008, doi: 10.1109/SURV.2008.080407.

[19] M. K. Marina and S. R. Das, "Ad hoc on-demand multipath distance vector routing," *Wirel. Commun. Mob. Comput.*, vol. 6, no. 7, pp. 969–988, 2006, doi: 10.1002/wcm.432.

[20] S. Deswal and S. Singh, "Implementation of Routing Security Aspects in AODV," *Int. J. Comput. Theory Eng.*, vol. 2, no. 1, pp. 135–138, 2010, doi: 10.7763/ijcte.2010.v2.129.

[21] A. A.Vani and D. S. Rao, "Providing of Secure Routing against Attacks in MANETs," *Int. J. Comput. Appl.*, vol. 24, no. 8, pp. 16–25, 2011, doi: 10.5120/2972-4003.

[22] P. M. Jawandhiya, M. Ghonge, M. S. Ali, and J. S. Deshpande, "A Survey of Mobile Ad Hoc Network Attacks," *SSRN Electron. J.*, vol. 2, no. 9, pp. 4063–4071, 2019, doi: 10.2139/ssrn.3451027.

[23] M. H. Rehmani, S. Doria, and M. R. Senouci, "A Tutorial on the Implementation of Ad-hoc On Demand Distance Vector (AODV) Protocol in Network Simulator (NS-2)," 2010, [Online]. Available: http://arxiv.org/abs/1007.4065.

[24] G. S. Ganpat Joshi, "A Novel Statistical Adhoc On-Demand Distance Vector Routing Protocol Technique is using for Preventing the Mobile Adhoc Network from Flooding

Attack," *Turkish J. Comput. Math. Educ.*, vol. 12, no. 6, pp. 1753–1765, 2021, doi: 10.17762/turcomat.v12i6.3779.

[25] A. K. Jain and A. Choorasiya, "Protocol in Mobile Ad Hoc Network," no. Icces, pp. 958–964, 2017.

[26] D. Sarkar, S. Choudhury, and A. Majumder, "Enhanced-Ant-AODV for optimal route selection in mobile ad-hoc network," *J. King Saud Univ. - Comput. Inf. Sci.*, 2018, doi: 10.1016/j.jksuci.2018.08.013.

[27] D. P. I. I. Ismail and M. H. F. Ja'Afar, "Mobile ad hoc network overview," *2007 Asia-Pacific Conf. Appl. Electromagn. Proceedings, APACE2007*, 2007, doi: 10.1109/APACE.2007.4603864.

[28] K. R. K. Reddy, "Improved Protocol Design with Security and QoS over MANET," vol. 3, no. 1, pp. 735–739, 2018.

[29] N. Thagele and P. Tripathi, "A Critical Review of AODV , DSDV and DSR Protocol Presence of Malicious Node in Mobile Ad-Hoc Network," vol. 3, no. 1, pp. 585–588, 2018.

[30] A. G. Chekol, "Priority Aware QoS Enhancement in AODV Routing Protocol in MANETs,Unpublished Master's Thesis, Department of Computer Science, Addis Ababa University," no. June 2018, 2018.

[31] T. Gebremichael, "Preventing Flooding Attack in MANETs using the reserved bits of AODV messages,Unpublished Master's Thesis, Department of Computer Science, Addis Ababa University," no. December, 2014.

[32] Bhagyalakshmi and A. K. Dogra, "Q-AODV: A Flood control Ad-Hoc on Demand

Distance Vector Routing Protocol," *ICSCCC 2018 - 1st Int. Conf. Secur. Cyber Comput. Commun.*, pp. 294–299, 2018, doi: 10.1109/ICSCCC.2018.8703220.

[33] K. Dheepan, "Security enhancement and certificate revocation in MANET using position and energy based monitoring," *Int. Arab J. Inf. Technol.*, vol. 16, no. 1, pp. 88–97, 2019.

[34] S. M. Zarei and R. Fotohi, "Defense against flooding attacks using probabilistic thresholds in the internet of things ecosystem," *Secur. Priv.*, vol. 4, no. 3, pp. 1–15, 2021, doi: 10.1002/spy2.152.

[35] L. D. Huy, L. T. Ngoc, and N. Van Tam, "AOMDV-OAM: A security routing protocol using OAM on mobile Ad Hoc network," *J. Commun.*, vol. 16, no. 3, pp. 104–110, 2021, doi: 10.12720/jcm.16.3.104-110.

[36] B. K. Rao and A. S. N. Chakravarthy, "Optimized AODV Routing Algorithm in MANET for Maximizing the Network Lifetime," no. 2, pp. 4054–4059, 2019, doi: 10.35940/ijrte.B3457.078219.

[37] L. N. Migiro, " Overview of MANET" *Thesis*, no. May, pp. 1–29, 2010.

[38] J. GROSREY and G. Neyer, "Analysis of MANET," *Demogr. Res.*, vol. 49, no. 0, pp. 1-33 : 29 pag texts + end notes, appendix, referen, 2003.

[39] A. K. Nayak, S. C. Rai, and R. Mall, *Introduction to NS2*. 2016.

[40] H. Vegda and N. Modi, "Review Paper on Mobile Ad-hoc Networks," *Int. J. Comput. Appl.*, vol. 179, no. 37, pp. 33–35, 2018, doi: 10.5120/ijca2018916859.

APPENDIX

A: Sample codes

```
#codes for calculating threshold value: prepared by hirpa
set
threshold[expr($n0+$n1+$n2+$n3+$n4+$n5+$n6+$n7+$n8+$n9+$n10+$n11+$n12+$n13+$n14
+$n15+$n16+$n17+$n18+$n19+$n20+$n21+$n22+$n23+$n24+$n25+$n26+$n27+$n28+$n29+
$n30+$n31+$
n32+$n33+$n34+$n35+$n36+$n37+$n38+$n39)/$NoN +$RATELIMIT]
#C++ codes for isolating and detecting rreq flooding attack
$ns at 0.0 "$ns trace-annotate \" threshold value is $threshold \""
#check the if condition: prepared by hirpa
if { $n0 > [expr $threshold] } {
# if condition is true then print the following
$ns at 0.0 "$ns trace-annotate \" n0 is attacker \""
} elseif { $n0 < [expr $threshold] } {
# if else if condition is true

$ns at 0.0 "$ns trace-annotate \"n0 is normal \""
} elseif { $n0 == [expr $threshold] } {
# if else if condition is true
$ns at 0.0 "$ns trace-annotate \" n0 is normal \""
}
if { $n1 > [expr $threshold] } {
# if condition is true then print the following
$ns at 4.0 "$ns trace-annotate \" n1 is attacker \""
} elseif { $n1 < [expr $threshold] } {
# if else if condition is true
$ns at 4.0 "$ns trace-annotate \"n1 is normal \""
} elseif { $n1 == [expr $threshold] } {
# if else if condition is true
$ns at 4.0 "$ns trace-annotate \" n1 is normal \""
}
```

```
if { $n2 > [expr $threshold] } {
# if condition is true then print the following
$ns at 0.0 "$ns trace-annotate \" n2 is attacker \""
} elseif { $n2 < [expr $threshold] } {
# if else if condition is true
$ns at 0.0 "$ns trace-annotate \"n2 is normal \""
} elseif { $n2 == [expr $threshold] } {
# if else if condition is true
$ns at 0.0 "$ns trace-annotate \" n2 is normal \""
}
if { $n3 > [expr $threshold] } {
# if condition is true then print the following
$ns at 0.0 "$ns trace-annotate \" n3 is attacker \""
} elseif { $n3 < [expr $threshold] } {
# if else if condition is true
$ns at 0.0 "$ns trace-annotate \"n3 is normal \""
} elseif { $n3 == [expr $threshold] } {
# if else if condition is true
$ns at 0.0 "$ns trace-annotate \" n3 is normal \""
}
if { $n4 > [expr $threshold] } {
# if condition is true then print the following
$ns at 3.0 "$ns trace-annotate \" n4 is attacker \""
} elseif { $n4 < [expr $threshold] } {
# if else if condition is true
$ns at 3.0 "$ns trace-annotate \"n4 is normal \""
} elseif { $n4 == [expr $threshold] } {
# if else if condition is true
$ns at 3.0 "$ns trace-annotate \" n4 is normal \""
}
if { $n5 > [expr $threshold] } {
```

```
# if condition is true then print the following
$ns at 2.0 "$ns trace-annotate \" n5 is attacker \""
} elseif { $n5 < [expr $threshold] } {
# if else if condition is true
$ns at 2.0 "$ns trace-annotate \"n5 is normal \""
} elseif { $n5 == [expr $threshold] } {
# if else if condition is true
$ns at 2.0 "$ns trace-annotate \" n5 is normal \""
}
if { $n6 > [expr $threshold] } {
# if condition is true then print the following
$ns at 1.0 "$ns trace-annotate \" n6 is attacker \""
} elseif { $n6 < [expr $threshold] } {
# if else if condition is true
$ns at 1.0 "$ns trace-annotate \"n6 is normal \""
} elseif { $n6 == [expr $threshold] } {
# if else if condition is true
$ns at 1.0 "$ns trace-annotate \" n6 is normal \""
}
if { $n7 > [expr $threshold] } {
# if condition is true then print the following
$ns at 2.5 "$ns trace-annotate \" n7 is attacker \""
} elseif { $n7 < [expr $threshold] } {

# if else if condition is true
$ns at 2.5 "$ns trace-annotate \"n7 is normal \""
} elseif { $n7 == [expr $threshold] } {
# if else if condition is true
$ns at 2.5 "$ns trace-annotate \" n7 is normal \""
}
if { $n8 > [expr $threshold] } {
```

```
# if condition is true then print the following
$ns at 2.05 "$ns trace-annotate \" n8 is attacker \""
} elseif { $n8 < [expr $threshold] } {
# if else if condition is true
$ns at 2.05 "$ns trace-annotate \"n8 is normal \""
} elseif { $n8 == [expr $threshold] } {
# if else if condition is true
$ns at 2.05 "$ns trace-annotate \" n8 is normal \""
}
if { $n9 > [expr $threshold] } {
# if condition is true then print the following
$ns at 4.0 "$ns trace-annotate \" n9 is attacker \""
} elseif { $n9 < [expr $threshold] } {
# if else if condition is true
$ns at 4.0 "$ns trace-annotate \"9 is normal \""
} elseif { $n9 == [expr $threshold] } {
```

… Up to node 40

Awk Scripts

#Average end-to-end delay

```
BEGIN {
max_packet_id = 0;
}



{
            action = $1;
time = $2;        node =
$3        type = $4;
application = $7;
packet_size = $8;
packet_id = $27;
if ((action == "s" ||
action == "r") && type
== "AGT" &&
application == "cbr") {
            packet_id =
strtonum(substr(packet_i
d,2,length(packet_id)-
2));                  if
(packet_id >
max_packet_id)
            max_packet_id =
packet_id;
            }



            if (action == "s" && type == "AGT" && application == "cbr")
start_time[packet_id] = time;
```

```
        if (action == "r" && type == "AGT" && application == "cbr") {

                stop_time[packet_id] = time;

        } else {
stop_time[packet_id] = -1;
        }

}



END {
delay = 0;
delay_n = 0;
        for (packet_id = 0; packet_id <= max_packet_id; packet_id++)
{               start = start_time[packet_id];                  stop =
stop_time[packet_id];                   packet_duration = stop - start;


                if (start < stop) {
delay += packet_duration;
                        delay_n++;

                }

        }

        print "Average End-to-End Delay    = "delay/delay_n "ms";

}
```

#residual energy

```
BEGIN {
```

```
initialenergy=90
maxenergy=0
n=20 node_id=999
}

{

# Trace line format: energy event = $1 time = $2 if (event
== "r" || event == "d" || event == "s"|| event== "f") { node_id
= $9 energy = $17
} if (event== "N")
{ node_id = $9
energy = $17
}

# Store remaining energy
finalenergy[node_id] = energy
}



END {

# Compute consumed energy for each node
for (i in finalenergy) { consumenergy[i] =
initialenergy-finalenergy[i] totalenergy +=
consumenergy[i] if(maxenergy <
consumenergy[i]){ maxenergy =
consumenergy[i] node_id = i
}

}
```

```
###compute average energy
averagenergy=totalenergy/n
####output
#for (i=0; i<n; i++) { #print("node",i,consumenergy[i])

#} printf("################ Designed by HIRPA GIRMA
###########################\n"); print("residual energy:",averagenergy) print("residual
energy in each nodes:",averagenergy/n) print("total energy:",totalenergy)
printf("########################################################################
####\n");



}
```

#PDR

```
# AWK Script for Packet Delivery Calculation for new Trace Format



BEGIN {

sent=0; received=0;
}




{   if($1=="s" &&
$4=="AGT")
   {
sent++;
   }

  else if($1=="r" && $4=="AGT")
```

```
   {
received++;
   }



}

END{  printf("################ Designed by HIRPA GIRMA
##########################\n");  printf " Packet Sent:%d",sent;  printf "\n Packet
Received:%d",received;  printf "\n Packets Dropped:%d", (sent-received);  printf "\n Packet
Delivery Ratio:%.2f\n",(received/sent * 100);
printf "=====Statistics Correct===== !! \n";
printf("##########################################################################\n");


}
```

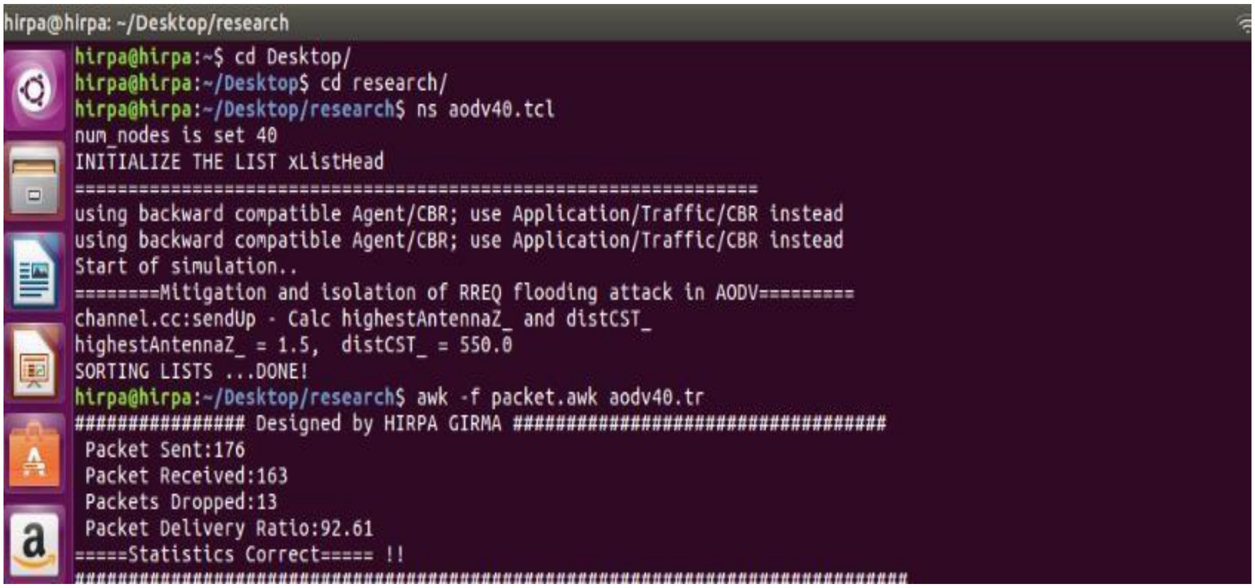
#Throughput

```
BEGIN {

       recvdSize = 0
startTime = 400
stopTime = 0
  }

      {             event =
$1              time = $2
node_id = $3
pkt_size = $8
level = $4
```

```
  # Store start time   if ((level == "AGT" || level == "IFQ") && (event == "s") &&
pkt_size >= 512) {     if (time < startTime) {             startTime = time
            }

       }



  # Update total received packets' size and store packets arrival time   if ((level ==
"AGT" || level == "IFQ") && (event == "r") && pkt_size >= 512) {        if (time
> stopTime) {              stopTime = time
            }

       # Rip off the header

       #hdr_size = pkt_size % 512

       #pkt_size -= hdr_size

       # Store received packet's size
recvdSize += pkt_size
       }

  }



  END {

       printf("Average Throughput[kbps] = %.2f\t\t
StartTime=%.2f\tStopTime=%.2f\n",(recvdSize/(stopTimestartTime))*(8/1000),startTime,stopTi
me)

  }
```

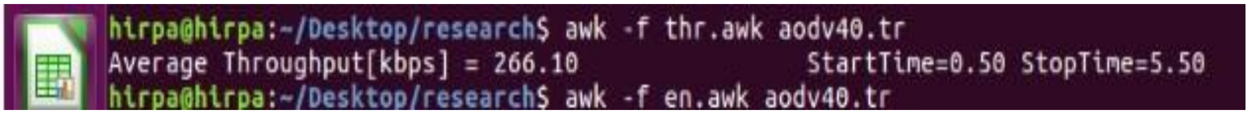
B. Sample screenshot outputs of AWK scripts over 40 nodes.

#sample screenshot of PDR



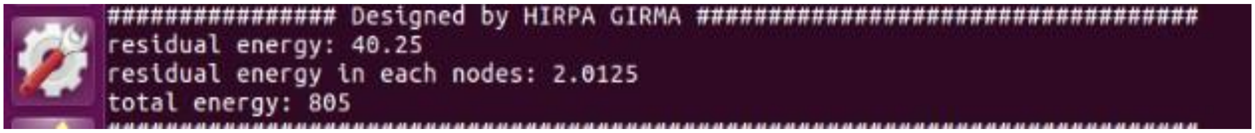
```
hirpa@hirpa: ~/Desktop/research
hirpa@hirpa:~$ cd Desktop/
hirpa@hirpa:~/Desktop$ cd research/
hirpa@hirpa:~/Desktop/research$ ns aodv40.tcl
num_nodes is set 40
INITIALIZE THE LIST xListHead
=================================================================
using backward compatible Agent/CBR; use Application/Traffic/CBR instead
using backward compatible Agent/CBR; use Application/Traffic/CBR instead
Start of simulation..
========Mitigation and isolation of RREQ flooding attack in AODV=========
channel.cc:sendUp - Calc highestAntennaZ_ and distCST_
highestAntennaZ_ = 1.5,  distCST_ = 550.0
SORTING LISTS ...DONE!
hirpa@hirpa:~/Desktop/research$ awk -f packet.awk aodv40.tr
############### Designed by HIRPA GIRMA ##################################
 Packet Sent:176
 Packet Received:163
 Packets Dropped:13
 Packet Delivery Ratio:92.61
=====Statistics Correct===== !!
################################################################################
```

#sample screenshot of Throughput



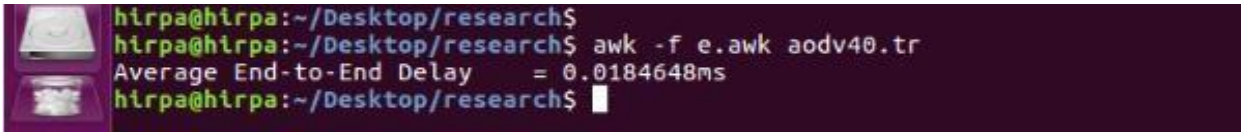
```
hirpa@hirpa:~/Desktop/research$ awk -f thr.awk aodv40.tr
Average Throughput[kbps] = 266.10              StartTime=0.50 StopTime=5.50
hirpa@hirpa:~/Desktop/research$ awk -f en.awk aodv40.tr
```

#sample screenshot of Residual energy



```
################ Designed by HIRPA GIRMA ###################################
residual energy: 40.25
residual energy in each nodes: 2.0125
total energy: 805
```

#sample screenshot of End to end delay



```
hirpa@hirpa:~/Desktop/research$
hirpa@hirpa:~/Desktop/research$ awk -f e.awk aodv40.tr
Average End-to-End Delay    = 0.0184648ms
hirpa@hirpa:~/Desktop/research$ █
```