Design and Development of an Efficient Secure AODV Routing Protocol

A Thesis submitted in partial fulfilment of the requirement for the award of the degree of

DOCTOR OF PHILOSOPHY

In

Computer Science and Engineering

Submitted by

Bhawna Singla

(Registration No. 950903037)

Under the Supervision of

Dr. A.K. Verma
Professor
Dept. of Computer Science and Engineering
Thapar Institute of Engineering and Technology, Patiala

Dr. L.R. Raheja
Professor (Retd.)
Indian Institute of Technology
Kharagpur (West Bengal)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING,
THAPAR INSTITUTE OF ENGINEERING AND TECHNOLOGY,
PATIALA-147004 (INDIA)

APRIL 2018

Certificate

I hereby certify that the work which is being presented in the thesis entitled " Design and Development of an Efficient Secure AODV Routing Protocol" in partial fulfillments of the requirements for the award of the degree of doctor of philosophy and submitted in the Department of Computer Science and Engineering at Thapar Institute of Engineering and Technology (Deemed University), Patiala is an authentic record of my own work carried during the period from January 2010 to April 2018 under the supervision of Dr. A.K. Verma, Professor, CSED, Thapar Institute of Engineering and Technology (Deemed University), Patiala and Dr. L.R. Raheja, Ex. Professor, IIT, Kharagpur.

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

Bhawna Singla

(Regn No. 950903037)

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

Date: 27th April 2018

Dr. A.K. Verma

Dr. L.R. Raheja

Abstract

Mobile Adhoc network (MANETs) applications are increasing every day due to inherent characteristics like infrastructure less nature, dynamic topology and multi-hop network. The path from source node to destination node is determined with the help of set of rules known as routing protocol. In MANETs, the routing protocols are divided into two major categories like proactive and reactive. This work discusses the Adhoc On-demand Distance Vector Routing Protocol (AODV), a kind of reactive routing protocol, for its study and modification. But MANETs is also vulnerable to number of attacks due to its nature. A lot of work has been done to make it more secure. In this work, the impacts of some of the popular attacks on a short term military rescue mission like MANETs scenario is studied. In particular, three attacks being studied on AODV routing protocol are a) Jellyfish attack b) Black hole attack c) Selfish node attack. To have the better analysis, the network size is varied firstly and then the number of malicious node is varied.

Detection of malicious nodes is a challenging task. Further, isolating such malicious nodes from communication is also a great challenge. The trust based on the packet forwarding behaviour of neighbor can be used for detecting misbehaviour as discussed in previous works in literature. This model has been already presented in several works. But, by the same trust based logic, some of the neighbors those who were silent and not actively participated in communications will get wrongly identified as malicious. So, simple trust based models will mark a lot of non-malicious nodes as malicious nodes. This will initiate a lot of link failures. That is, the link between sources to destination will get broken at different locations on their path because of this false identification of malicious nodes.

In this work, periodic trust handshake based detection (PTH-AODV) of black hole attack is proposed. PTH-AODV using NS2 is implemented and its performance is compared with the results of AODV and AODV under attack. In this work, further dynamic trust handshake based detection (DTH-AODV) of black hole attack is proposed. DTH-AODV using NS2 is implemented and its performance is compared with the results of previous PTH-AODV, AODV and AODV under jellyfish attack, black hole attack and selfish node attack.

The main advantage of the proposed PTH-AODV and DTH-AODV is: they will detect and prevent the malicious nodes in the very early stage of route discovery process. So, they will not need any manipulation in routing tables in the route resolving process, because, by the design, it will avoid including malicious hops in routing table of normal nodes at the route discovery process itself. According to the arrived results, proposed trust handshake based malicious node detection and prevention mechanism worked good and successfully detected black hole nodes in the network and avoided establishing routes though them. The proposed DTH-AODV improved the throughput whereas the PDF is almost equal to that of AODV and the overall performance of DTH-AODV was observed to be better than previously proposed PTH-AODV.

Keywords: MANETs, AODV, Secure Routing Protocols, PTH-AODV, DTH-AODV

Acknowledgement

I would like to express my gratitude to my guides Dr. Anil Kumar Verma and Dr. L.R. Raheja. They have guided and supervised me in my pursuit of knowledge which has come up in the form of the present work. I thank them for encouraging me at every step which helped me to complete this task. Their advice on both researches as well as on my career has been priceless. I express my deep regards to Dr. P. Gopalan, Director, TU for all the facilities provided to me during my research work. I would also like to thank my committee members Dr. Maninder Singh, Chairman, CSED and Dr. Rajesh Khanna and Dr. V.P. Singh, Member, PhD committee and Dr. O. P. Pandey, Dean of Research & Sponsored Projects, for their meaningful comments and useful suggestions during my research which helped me a lot in improving my shortcomings.

I will also like to thank my parents for their support, their blessings and prayers for me that gave me strength throughout this long journey.

Mere words cannot express my feeling of debt and gratitude to my son Kanan Gupta and daughter Prisha Gupta for allowing me the time which I had to devote to my work meant for them. At the end I would like to express appreciation for my husband Anish Gupta, without whom this work would not have been possible.

Bhawna Singla

Table of Contents

Candidate's Declaration i

Abstract ii

Acknowledgements iv

Table of Contents v

List of Abbreviations viii

List of Tables x

List of Figures xii

List of Publications xvi

Chapter 1: Introduction

1.1 Mobile Adhoc Network (MANETs) 1

1.2 MANETs Characteristics 2

1.3 MANETs Routing Protocol 3

1.3.1 Proactive Routing Protocols 3

1.3.2 Reactive Routing Protocols 5

1.3.3 Hybrid Routing Protocols 6

1.3.4 Positional aided Routing Protocols 7

1.4 Performance Metric of Routing Protocol 9

1.5 Adhoc On-demand Distance Vector Routing Protocol(AODV) 11

1.5.1 Control Messages 12

1.5.2 Phases of AODV Protocol 13

1.6 Research Gaps 18

1.7 Problem formulation 18

1.8 Research Objectives 19

1.9 Methodology 19

1.10 Our Contributions 20

1.11 Thesis Organization 21

Chapter 2: Literature Review

2.1 Vulnerabilities of AODV 24

2.2 Classification of attacks 25

2.3 Common Attacks in MANETs 29
2.4 Attacks on AODV Routing Protocol 31
2.4.1. Jellyfish Attack 32
2.4.2 Selfish node Attack 33
2.4.3 Black hole Attack 34
2.5 Existing Security Enhancements to Routing Protocol 36
2.5.1 Cryptographic Based Routing 36
2.5.2 Reputation Based Routing 42
2.5.3 Embedding Trust Metric into Routing Protocol 45
2.6 Summary 51

Chapter 3: Simulation Setup, Comparison of Attacks

3.1 NS2.35 Simulator 56
3.1.1 Nodes, Links and Packet Forwarding 57
3.1.2 Agents 58
3.1.3 Mobile Networking in NS 59
3.1.4 NAM 60
3.2 Simulation of Attacks under NS2 Simulator 60
3.2.1 Pseudo Code of Attacks 60
3.2.2 Methodology 63
3.2.3 Simulation Parameters 64
3.3 Results of Attacks under NS2 Simulator 66
3.3.1 Analysis-I Network Size Vs Performance 66
3.3.2 Analysis-II Malicious Nodes Vs Performance 73
3.3 Summary 80

Chapter 4: Periodic Trust Handshake based Malicious Behaviour Detection Mechanisms

4.1 Periodic Trust Handshake Based Malicious Behaviour Detection Mechanism 82
4.1.1 Characteristics of Trust Metric 83
4.1.2 The main changes that are made in basic Trust based AODV 84

4.1.3 Implementation of Periodic Trust Handshake Mechanism 85
4.2 Simulation Parameters 90
4.3 Results of PTH-AODV 91
4.3.1 Analysis of Results with respect to Different Network Size 91
4.4 Comparison of PTH-AODV with existing secure routing protocol 99
4.4 Summary 100

Chapter 5: Dynamic Trust Handshake based Malicious Behaviour Detection Mechanisms

5.1 Dynamic Trust Handshake Based Malicious Behaviour Detection Mechanism 102
5.1.1 Implementation of Dynamic Trust Handshake Mechanism 102
5.2 Results of DTH-AODV 108
5.2.1 Analysis of Results with respect to Different Network Size 108
5.3 Comparison of performance of PTH-AODV and DTH-AODV with Selfish node attack and Black hole attack 117
5.3.1 Analysis of Results with respect to Different Network Size 117
5.4 Comparison of PTH-AODV and DTH-AODV with other Trust Based Routing Algorithm 129
5.5 Summary 133

Chapter 6: Conclusion And Future Scope 135

References 140

List of Important Abbreviations

Abbreviations/Acronyms	Description
AODV	Adhoc On-demand Distance Vector routing
ABR	Associativity Based Routing
ACR	Ant-Colony based Routing Algorithm
ARAN	Authenticated Routing for Adhoc Network
BISS	Building secure routing out of an Incomplete Set of Security associations
CBR	Cluster Based Routing protocol
CGSR	Cluster-head Gateway Switching Routing
CONFIDANT	Cooperation Of Nodes: Fairness In Dynamic Ad-hoc NeTworks
CORE	COoperation enforcement based On REputation
DDR	Distributed Dynamic Routing
Dos	Denial Of Service attack
DREAM	Distance Routing Effect Algorithm for Mobility
DSDV	Destination Sequenced Distance Vector
DSR	Dynamic State Routing
DST	Distributed Spanning Tree based routing protocol
DTH-AODV	Dynamic Trust Handshake based malicious behaviour detection mechanisms based on AODV
EED	End-to-End Delay
EMPSO	Energy-aware MultiPath routing Scheme based on particle swarm Optimization
FSR	Fisheye State Routing
GSR	Global State Routing
HOPNET	Hybrid ant colony OPtimization routing algorithm for mobile ad hoc NETwork
LAR	Location Aided Routing
MAC	Media Access Control
MANETs	Mobile Adhoc NETworks
NS2	Network Simulator version 2
Oantalg	an Orientation based ANT colony ALGorithm for mobile ad hoc networks

PTH-AODV	Periodic Trust Handshake based malicious behaviour detection mechanisms based on AODV
QID	random Query Identifier
QSEQ	Query SEQuence number
RDP	Route Discovery Packet
RREQ	Route REQuest control packet
RREP	Route REPly control packet
RERR	Route ERRor control packet
RREP_ACK	Route REPly Acknowledgment
SAODV	Secure AODV
SAR	Security Aware adhoc Routing
SEAD	Secure Efficient AODV
SLSP	Secure Link State routing Protocol
SPC	Shortest Path Confirmation
SRP	Secure Routing Protocol
SYN	Signal sent by client
SYN_ACK	Acknowledgment sent by server
TBOR	Tree Based Opportunistic Routing for mobile ad hoc networks
TBRPF	Topology Broadcast Reverse Path Forwarding
TCL/OTCL	Tool Command Language in NS2/ Object oriented extension of TCL
TCP	Transmission Control Protocol
TIARA	Techniques for Intrusion Resistant Adhoc Routing Algorithm
TORA	Temporary Ordered Routing Algorithm
TTP	Trusted Third Party
WRP	Wireless Routing Protocol
ZHLS	Zone based Hierarchical Link Structured
ZRP	Zone Routing Protocol

List of Tables

Table number	Description	Page number
2.1	Security attack classification 1	26
2.2	Summary of cryptographic routing protocol	39
2.3	Defence against attacks	40
2.4	Comparison of existing and common reputation based scheme	44
2.5	Existing weighted trust based models	48
3.1	Comparison of various simulation tools	54
3.2	Simulation parameters	65
3.3	Parameters for setting up TCP flows	65
3.4	Parameters for analyzing the impacts of attacks	66
3.5	Trace file of AODV without attack	66
3.6	Trace file of AODV with Jellyfish reorder attack	67
3.7	Trace file of AODV with Jellyfish periodic dropping attack	67
3.8	Trace file of AODV with Jellyfish delay variance attack	67
3.9	Trace file of AODV with Selfish node attack	67
3.10	Trace file of AODV with Black hole attack	68
3.11	Trace file of AODV with Jellyfishreorderattack and variable malicious node	73
3.12	Trace file of AODV with Jellyfish periodic dropping attack and variable malicious node	74
3.13	Trace file of AODV with Jellyfish delay variance attack and variable malicious node	74
3.14	Trace file of AODV with Selfish node attack and variable malicious node	74
3.15	Trace file of AODV with Black hole attack and variable malicious node	74
4.1	Parameters values of network in NS2	90
4.2	Total number of nodes, number of malicious node and different attack scenarios	91
5.1	Trace file of modified AODV with variable nodes	117
5.2	Trace file of AODV with Black hole attack	117
5.3	Trace file of AODV with Selfish node attack	118

5.4	Trace file of DTH-AODV with Black hole attack	118
5.5	Trace file of DTH-AODV with Selfish node attack	118
5.6	Trace file of PTH-AODV with Black hole attack	118
5.7	Trace file of PTH-AODV with Selfish node attack	119
5.8	Comparison of characteristics of existing AODV based trust routing protocol with PTH-AODV and DTH-AODV	130

List of Figures

Figure number	Description	Page number
1.1	Category of wireless local area network	1
1.2	Infrastructure based approach	2
1.3	Infrastructure less approach	2
1.4 (a)	Various categories of routing protocols (1999-2003): Proactive , Reactive, Hybrid and Position aided routing protocol	8
1.4 (b)	Various categories of routing protocols (2003-2018): Proactive , Reactive, Hybrid and Position aided routing protocol	9
1.5	Format of RREQ (RFC 3561)	12
1.6	Format of RREP (RFC 3561)	12
1.7	Format of RERR (RFC 3561)	13
1.8	Flowchart showing the generation of RREQ control packet	14
1.9	Flowchart showing the forwarding of RREQ	15
1.10	Flowchart showing generation of route reply	16
1.11	Routing in AODV with example	17
1.12	Modeling of proposed work	21
2.1	Attack Classification	25
2.2	Layer wise description of attacks	28
2.3	Black hole attack on the network where 1 is the source node, 4 is the destination node and 6 is the black hole node.	29
2.4	Wormhole attack in AODV routing protocol. Here, 1 is the source node and 7 is the destination node.	30
2.5	Example showing the working of selfish node attack in a network consisting of 7 nodes where 1 is the source node and 7 is the destination node and X is the selfish node	33
2.6	Example showing the working of black hole attack where 1 is the source node, 3 is the destination node and 4 is the malicious node.	34
2.7	Flowchart showing the working of Black hole attack	35
2.8 (a)	Cryptographic solution of adding security in routing protocols (1999-2007)	41

2.8 (b)	Cryptographic solution of adding security in routing protocols (2008-2018)	42
2.9	Properties of trust metric	46
2.10	Various methods of trust computation	48
3.1	Various common tools for simulation of algorithms	57
3.2	NS2 block diagram	58
3.3	NS2 components	60
3.4	Various fields of trace file	60
3.5	Pseudo code of Forward packet function	61
3.6	Pseudo code of OnRecieveRReq Function	63
3.7	Comparison of Attackers Vs Sent Data Packets in presence of attacks on AODV	68
3.8	Comparison of Attackers Vs Received Data Packets in presence of attacks on AODV	69
3.9	Comparison of Attackers Vs Dropped at The application layer in presence of attacks on AODV	70
3.10	Comparison of Attackers Vs Maliciously Dropped at Routing Layer in presence of attacks on AODV	70
3.11	Comparison of Attackers Vs Throughput in presence of attacks on AODV	71
3.12	Comparison of Attackers Vs Dropped PDF in presence of attacks on AODV	71
3.13	Comparison of Attackers Vs End-to-end Delay in presence of attacks on AODV	72
3.14	Comparison of Attackers Vs Battery Energy in presence of attacks on AODV	73
3.15	Comparison of Network Size Vs Sent Packets on AODV with varying malicious node	75
3.16	Comparison of Network Size Vs Received Packets on AODV with varying malicious node	76
3.17	Comparison of Network Size Vs Packets Dropped At The application layer on AODV with varying malicious node	77
3.18	Comparison of Network Size Vs Malicious Drops at Routing Layer on AODV with varying malicious node	77
3.19	Comparison of Network Size Vs Throughput on AODV with varying malicious node	78
3.20	Comparison of Network Size Vs PDF on AODV with varying malicious node	78
3.21	Comparison of Network Size Vs End-to-end Delay on AODV with varying malicious node	79
3.22	Comparison of Network Size Vs Battery Energy on AODV with varying malicious node	80
4.1	Figure showing calculation of trust values	82

4.2	Calculation of malicious node	82
4.3	Removal of unwanted neighbour	84
4.4	Calculation of Trust value	85
4.5	The Periodic trust handshake message handler	86
4.6	The process flow of periodic trust handshake based malicious node detection and prevention in AODV	87
4.7	The pseudo code of PTH-AODV	88
4.8	Comparison of Network Size Vs Sent Packets in PTH-AODV	92
4.9	Comparison of Network Size Vs Received Packets in PTH-AODV	92
4.10	Comparison of Network Size Vs Routing Load in PTH-AODV	93
4.11	Comparison of Network Size Vs MAC Load in PTH-AODV	93
4.12	Comparison of Network Size Vs Packets Dropped At The application layer in PTH-AODV	94
4.13	Comparison of Network Size Vs Throughput in PTH-AODV	95
4.14	Comparison of Network Size Vs PDF in PTH-AODV	95
4.15	Comparison of Network Size Vs End-to-end Delay in PTH-AODV	96
4.16	Comparison of Network Size Vs Battery Energy in PTH-AODV	97
4.17	Comparison of Network Size Vs Overhead in PTH-AODV	98
4.18	Comparison of Network Size Vs MAC Layer Dropped in PTH-AODV	98
4.19	Comparison of Network Size Vs Malicious Drops at Routing Layer in PTH-AODV	99
4.20	Comparison of various routing protocol by varying the Pause time	100
5.1	Dynamic trust handshake mechanism	103
5.2	Process flow of dynamic trust handshake mechanism	105
5.3	The pseudo code of DTH-AODV	106
5.4	Comparison of Network Size Vs Sent Packets in DTH-AODV	109
5.5	Comparison of Network Size Vs Received Packets in DTH-AODV	109
5.6	Comparison of Network Size Vs Routing Load in DTH-AODV	110
5.7	Comparison of Network Size Vs MAC Load in DTH-AODV	111

5.8	Comparison of Network Size Vs Packets Dropped At The application layer in DTH-AODV	111
5.9	Comparison of Network Size Vs Throughput in DTH-AODV	112
5.10	Comparison of Network Size Vs PDF in DTH-AODV	112
5.11	Comparison of Network Size Vs End-to-end Delay in DTH-AODV	113
5.12	Comparison of Network Size Vs Battery Energy in DTH-AODV	114
5.13	Comparison of Network Size Vs Overhead in DTH-AODV	115
5.14	Comparison of Network Size Vs MAC Layer Dropped in DTH-AODV	116
5.15	Comparison of Network Size Vs Malicious Drops at Routing Layer in DTH-AODV	116
5.16	Comparison of Network Size Vs Sent Packets in PTH-AODV and DTH-AODV	120
5.17	Comparison of Network Size Vs Received Packets in PTH-AODV and DTH-AODV	120
5.18	Comparison of Network Size Vs Routing Load in PTH-AODV and DTH-AODV	121
5.19	Comparison of Network Size Vs MAC Load in PTH-AODV and DTH-AODV	121
5.20	Comparison of Network Size Vs Packets Dropped At The application layer in PTH-AODV and DTH-AODV	122
5.21	Comparison of Network Size Vs Throughput in PTH-AODV and DTH-AODV	123
5.22	Comparison of Network Size Vs PDF in PTH-AODV and DTH-AODV	123
5.23	Comparison of Network Size Vs End-to-end Delay in PTH-AODV and DTH-AODV	125
5.24	Comparison of Network Size Vs Battery Energy in PTH-AODV and DTH-AODV.	126
5.25	Comparison of Network Size Vs Overhead in PTH-AODV and DTH-AODV.	126
5.26	Comparison of Network Size Vs MAC Layer Dropped in PTH-AODV and DTH-AODV.	127
5.27	Comparison of Network Size Vs Malicious Drops at Routing Layer in PTH-AODV and DTH-AODV.	128

List of Publications

Papers published in Refereed Journals:

1. Bhawna Singla, A. K. Verma and L. R. Raheja, "**An Evaluation on Selfish node Attack and Jellyfish Attacks Under AODV Routing Protocol"** International Journal in Foundations of Computer Science & Technology March 2017, Volume 7, Number 2, p.p 15-28.

2. Bhawna Singla, A. K. Verma and L. R. Raheja, "**A Comparative Analysis of Jellyfish Attacks and Black hole Attack with Selfish node Attack under AODV Routing Protocol"** British Journal of Mathematics and Computer Science 2017, Volume 21, p.p.1-18.

3. Bhawna Singla, A. K. Verma and L. R. Raheja, " **Performance Analysis of AODV in Presence of Attacks** " WSEAS Transactions on Communication 2017, Volume 6, p.p.85-93 (Scopus indexed IF = 0.7).

Papers published/accepted in Book as Chapter:

4. Bhawna Singla, A. K. Verma and L. R. Raheja, "**Preventing Black hole Attack in AODV Routing Protocol using Dynamic Trust Handshake Based Malicious Behaviour Detection Mechanism"** Machine Learning for Computer and Cyber Security: Principles, Algorithms, and Practices (Accepted).

Papers published in International Conference Proceedings:

5. Bhawna Singla, A. K. Verma and L. R. Raheja, "**Simulation of AODV under different attacks**" Proceedings of International conference of Emerging Technologies-2014 pp.79-86 , NCCE, Panipat.

Papers published in National Conference Proceedings:

6. Bhawna Singla, A. K. Verma and L. R. Raheja, "**Performance Analysis of AODV in Presence of Black hole Attack** " Proceedings of the National Conference on Advancements in the Era of Multi Disciplinary Systems (AEMDS-2013) (Elsevier Publications 2013) p.p 241-244 TERRI, Kurukshetra.

Papers under Review

7. Bhawna Singla, A. K. Verma and L. R. Raheja, "**Preventing Black hole attacks in AODV routing protocols using periodic trust handshake based malicious behaviour detection system**" IGI journal of International Journal of Information Security and Privacy (IJISP) (ESCI indexed).

Chapter-1

Introduction

Mobile Adhoc NETworks (MANETs) is one of the emerging fields that have seamless applications in the field of emergency situations like rescue operations, commercial applications like virtual classrooms, medical - like disease diagnosis etc. Mobile adhoc networks have vast applications because of its characteristics like wireless communication medium, minimal cost, accessible to more than one technology. However, security is one of the major concerns as these technologies require secure routing protocols.

1.1 Mobile Adhoc NETworkS (MANETs)

Mobile Adhoc NETworkS (MANETs) [1,2] is a collection of three terms Mobile, Adhoc and Network. It means that it is collection of nodes or stations that are connected over a wireless media where nodes have the flexibility to roam here and there thus to make dynamically and temporary topology. The most common wireless technology used now days is Wireless Local Area Network (WLAN). The WLAN has a communication range of 100-500m that means it is operational inside a building or a cluster of buildings. The WLAN can be implemented in two ways: (a) infrastructure based approach (b) infrastructure less approach (Adhoc) a shown in figure 1.1.

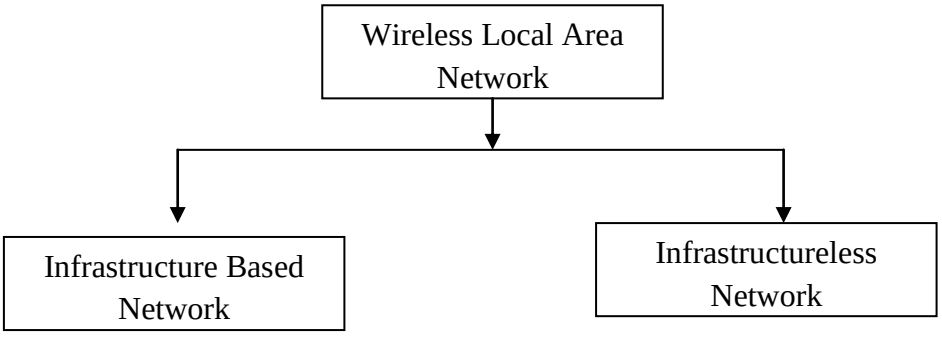

Figure 1.1: Category of wireless local area network

a) Infrastructure based approach [3]: In this approach, all nodes are divided into number of cells and each cell has a centralized controller called Access point as shown in figure 1.2. The function of the access point is that it is connected to a wired network for internet connectivity.

b) Infrastructureless approach [4]: Here, a Peer-to-Peer network is formed between each node that is in each other's range as shown in figure 1.3. Nodes keep on moving randomly so as make a temporary network. It does not require any fixed centralized controller as in infrastructure based network. Currently, IEEE 802.11 standard is used for implementing Adhoc networks.

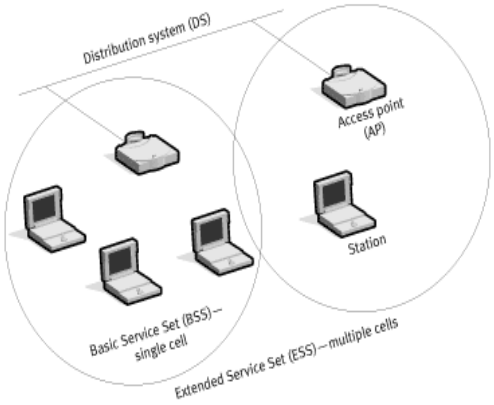

Figure1.1: Infrastructure based approach [3]

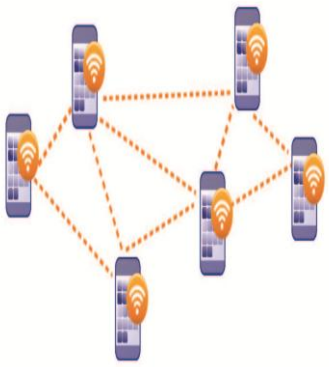

Figure 1.3: Infrastructure less approach [4]

MANETs Operating Principal: If a node transmits a packet to node that is in communication range of each other, then this communication is known as single hop transmission but if the node is not in communication range then sender node first transmits a packet to node which is in range (neighbor node), the neighbor node further transmits the packet to its neighbor till the packet is reached to the destined node. The intermediate node is known as the router and the communication is known as the multi-hop communication. To avoid the packet from traversing an indefinite loop path, a loop-free path is chosen which is known as a route.

1.2 MANETs Characteristics

Mobile Adhoc networks have certain characteristics that make them unique and versatile to a number of applications.

a) Communication Channel: MANETs uses wireless transmission media with limited transmission range.

b) No Centralized Infrastructure: There is no requirement of any centralized infrastructure such as base station, which makes them suitable for rescue applications such as the military. Moreover, MANETs do not work under the constraint of fixed infrastructure which also makes them fault resilient in a number of situations.

c) Mobile Nodes: All the nodes are free to roam here and there. The addition and deletion of nodes occur just by the interaction among the nodes.

d) Multi-hop Routing: All the nodes have limited transmission range. If the node wants to communicate with other nodes that is in direct transmission range then it can do so directly. But if the other node is not in direct transmission range then sender node first transmits to the neighbor node. The neighbor node further transmits to next neighbor node until it reaches to the destination nodes.

e) Dynamic Topology: As the nodes are mobile, so the topology is dynamic in nature that keeps on changing as per the movement of node.

1.3 MANETs Routing Protocols

Routing protocol [5-9] is defined as a set of rules that determine the path from source to destination. The routing protocols of MANETs can be classified into two categories: proactive and reactive protocols. Another category of routing protocol called hybrid protocol is also proposed that combines benefits of both reactive and proactive protocols. Still, another category called position aided routing protocol is also defined in some research papers based on the geographical distance of the nodes.

1.3.1 Proactive Routing Protocols: In proactive routing protocol, every node maintains the information regarding the path to other nodes. This is done by exchanging topological information in the form of the routing table. This information is updated periodically by nodes so as to determine the latest path among the nodes. Therefore when a source node

wants to transmit data to the destination node, it knows the routing path in advance. Some examples of proactive routing protocols are as shown below.

a) **Destination Sequenced Distance Vector Routing (DSDV)** [10]: It contains one routing table per node consisting of reachable destinations and number of hops to reach to those destinations. The shortest path is calculated by Bellman-Ford algorithm. To determine the freshness of route, the sequence number is assigned by the destination node. To reduce the network traffic routing updates are broadcast by two types of messages: Full damp and incremental messages.

b) **Wireless Routing Protocol (WRP)** [11]: In this Protocol, each node maintains four tables: Distance table, Routing table, Link-cost table, Message retransmission list (MRL) table containing the sequence number of update messages and acknowledgments. All the updates are broadcasted to the immediate neighbors so as to reduce the traffic.

c) **Global State Routing (GSR)** [15]: This is link-state based routing where each node maintains the link state table and exchanges the link state information with its neighbors at regular intervals. As this is link-state based routing, therefore routing overhead is very high.

d) **Fisheye State Routing** [16]: It is extension to GSR in which the information is updated frequently with the closer nodes than the faraway nodes. This routing is also link-state based routing but it reduces the routing overhead by ignoring faraway nodes in comparison of closer nodes.

e) **Cluster-head Gateway Switching Routing (CGSR)** [17]: Here nodes are grouped into clusters and the routing overhead is reduced as the routing messages are not flooded rather they are sent to clusters only.

f) **Optimized Link State Routing (OLSR)** [18]: It aims to minimize the size of update message and number of rebroadcasting nodes. It uses Multipoint replacing strategy (MPR). In this strategy, each node maintains the list of nodes called the relay node. Any node from the relay set can only retransmit the update messages while the other node can only read or process that packet.

g) **Topology Broadcast reverse path forwarding (TBRPF)** [18]: it uses link state routing technique that performs routing hop by hop. It uses reverse path forwarding (RPF) i.e. the update messages are also transmitted in reverse path in spanning tree.

1.3.2 Reactive Routing Protocol [19, 20, 21]: Reactive routing protocol is also called On-Demand routing protocol. In this protocol, route is discovered whenever it is needed instead of that each node maintains the route to all other nodes as in proactive routing protocol. Examples include Dynamic state routing (DSR)[12], Adhoc on-demand distance vector (AODV)[13], Temporary ordered routing algorithm (TORA)[14] etc. Two main phases of communication in on-demand routing protocol are:

- **Route Discovery:** Whenever a source node wants to transmit data to the destination node and route is not present, it initiates route discovery process. Source nodes check for the available route from source to destination. The route discovery process determines the path to the destination node (identified by destination IP address) including intermediate node.
- **Route Maintenance:** Due to consistently changing topology, the network may face frequent link failures. So, route maintenance is necessary. The acknowledgment mechanism in reactive routing protocols helps in route maintenance.

a) **Adhoc On-demand Distance Vector Routing (AODV)** [13]: It is based on DSDV but the routes are sent only upon requirement. Each node carries information regarding the destination address and sequence number.

b) **Dynamic Source Routing (DSR)** [12]: In this routing, each packet carries complete path information from source to destination. It has the advantage that every node stores multiple routes for the same source and destination which makes it more reliable but it also makes it unsuitable for large and highly dynamic network due to high overhead.

c) **Temporally Ordered Routing Algorithm (TORA)** [14]: This routing protocol uses link reversal and link repair mechanism. It also creates the DAGs for the purpose of finding optimized routes. It uses flooding technique to determine the route and nodes maintain multiple routes to the destination to ensure reliability.

d) **Associativity Based Routing (ABR)** [22]: This is reactive routing protocol where more stable routes are given more preference over the less stable route. Each node maintains the associativity table for each node. The entry is marked 1 whenever a

route is found. The route selection process prefers the node with higher associativity tick.

e) **Signal Stability Adaptive (SSA)** [23]: It is an improvement over ABR where the route is chosen based on signal strength and stability. As in ABR, it may result in shortest route however the route will stay longer. One disadvantage of SSA is that latency time of route is higher as the intermediate nodes are not allowed to respond to the repond to the route request.

f) **Ant-Colony based Routing Algorithm (ACR)** [24]: It uses food searching technique of ants in the way that they use pheromones to follow each other. Some other advancements to ant colony based routing algorithms are as shown in figure 1.4 (b): A hybrid ant colony optimization routing algorithm for mobile ad hoc network (HOPNET)[25], An orientation based ant colony algorithm for mobile ad hoc networks (Oantalg)[26], tree based opportunistic routing for mobile ad hoc networks (TBOR)[27], Energy-Aware Multipath Routing Scheme Based on Particle Swarm Optimization (EMPSO)[28].

g) **Cluster based routing protocol** [29]: In this, all the nodes are divided into clusters with one cluster head. The routing information is exchanged with the only cluster to avoid routing overhead.

1.3.3 Hybrid Routing Protocol: It is a combination of proactive and reactive routing protocol i.e it reduces the excessive overhead of proactive routing protocol by restricting the localized route update to on-demand. In this protocol, nodes with similar behavior make the backbone of the network so that they there is a route between them are less dynamic and they are determined in advance by using the proactive routing protocol. Further, a faraway node uses the common reactive routing protocol to find the route. In this way, the routing overhead to establish route every time is avoided.

a) **Zone Routing Protocol (ZRP)** [30]: It divides the protocol in zones so that the route within a zone is determined by proactive routing protocol and routes outside the zones are determined reactively.

b) **Zone based Hierarchical Link structured (ZHLS)** [31]: In this protocol, zones are hierarchically structured. If the destination node belongs to the same zone than

intrazone processing is done otherwise packets are sent to interzone which are optimized by using the hierarchies of zones.

c) **Distributed Spanning Tree Based Routing Protocol (DST)** [32]: is an example of tree based routing protocol where the nodes are structured into a tree. The update messages are sent from source node along the edges of the tree.

d) **Distributed Dynamic Routing (DDR)** [33] is an extension to DST but it does not require the root node.

1.3.4 Positional aided Routing Protocol: These routing protocols utilize the geographical information of protocols to make the routing decisions.

a) **Distance Routing Effect Algorithm for Mobility (DREAM)** [34]: Here, each node utilizes the GPS system to determine its geographical coordinates thereby maintaining the location table. The routing overhead is further reduced by broadcasting the routing message proportional to mobility. i.e. the stationary node does not send any update message while more mobile node sends more message.

b) **Location Aided Routing (LAR)** [35]: It is flooding based routing such as DSR but uses the location information to avoid overhead.

As routes are determined in advance in proactive routing protocol while they are determined whenever required in the reactive routing protocol. Therefore, Reactive routing protocols incorporate more delays in route determination mechanism while reactive routing protocols decrease the overhead to maintain the path among all the nodes in the network. Therefore, these protocols are suitable for situations where low routing overhead is required. However, the hybrid routing protocols and position aided routing protocols are more expensive as compared to the other two [36-38]. For research in this work, Adhoc On-demand Distance Vector (AODV), discussed in section 1.5, is chosen. Some most commonly referred types of routing protocols are summarized in figure 1.4 and are discussed here.

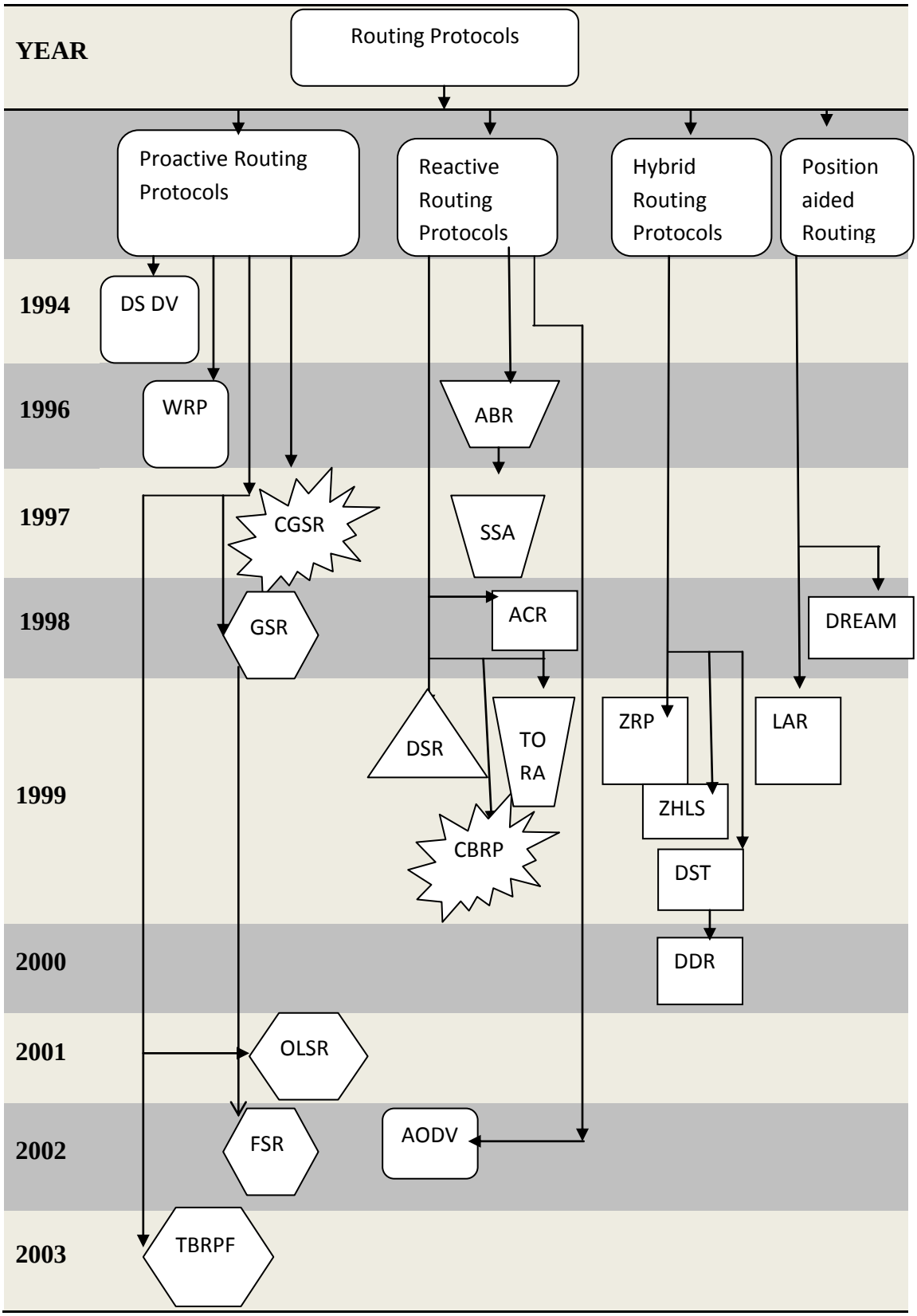

Figure 1.4 (a) shows the various categories of routing protocols: Proactive , Reactive, Hybrid and Position aided routing protocol. The distance vector based algorithms are shown in ▭, link state algorithms are shown in⬡ , cluster based algorithms are shown in ✸ , source routing is shown in △ and combination of link state and distance vector is shown in ▭ .

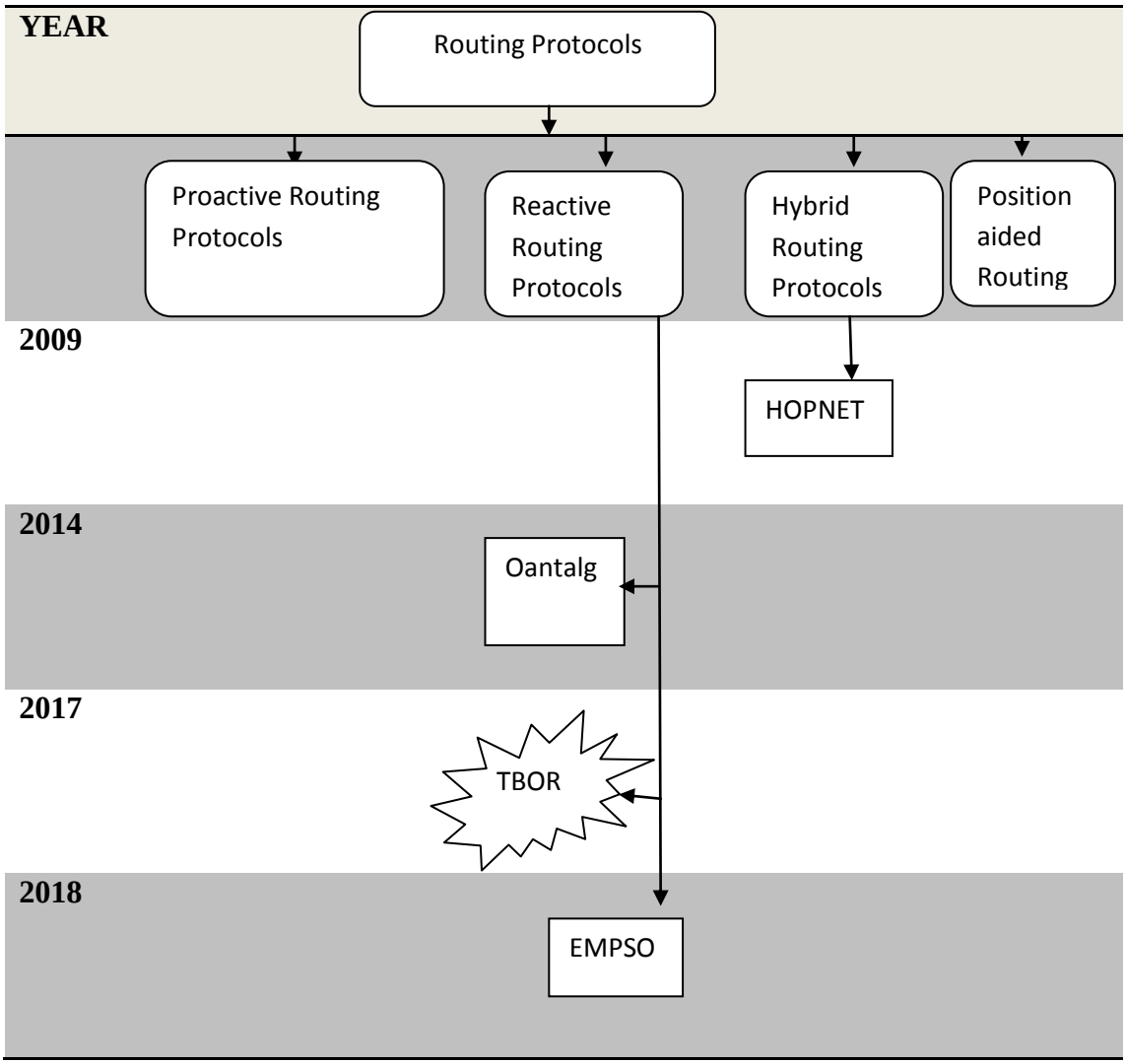

Figure 1.4 (b) shows the various categories of routing protocols (2003-2018): Proactive, Reactive, Hybrid and Position aided routing protocol.

1.4 Performance Metric of Routing Protocol

The performance of Routing Protocol can be compared in terms of the following metrics [39,40].

a) Cumulative sum of the number of packets sent, received or dropped: The cumulative sum is the sequence of partial sums of a given sequence where a partial sum of first N terms in a sequence $(a_k)^n{}_{k=1}$ is given by

$$S_N = \sum_{k=1}^{N} a_k \quad (1.1)$$

For example, cumulative sum of the sequence {a,b,c,…..} are a, a+b, a+b+c,….. In a simulation, the cumulative sum of the dropped packet at all nodes and the cumulative sum of the number of packet at the malicious or selfish node is

considered so as to see the percentage of harm it makes to the current routing protocol.

b) **Throughput of packet**: Throughput is the ratio of the delivered packet to the destination per unit of time which can be expressed as

$$T = \frac{P_r / P_s}{T} \quad (1.2)$$

Where P_r is the total number of received packet at the destination, P_s is the packet sent by the source and T is the time taken. Greater the value of the throughput means better the performance of the protocol. In this paper, the throughput of sent and dropped packets are compared. It is measured in kilobytes per second (Kbps).Another variant of throughput is PDF which is the ratio of the total number of packets received at the destination to the total number of packets sent.

c) **End-to-end delay (EED)**: Data packets in the network are not immediately received at the destination. The delay between the time at which data packet is generated and received at the destination is known as end-to-end delay (EED). EED may be due to route discovery process or it may be due to waiting queue generated during packet transmission. Only data packets successfully delivered to the destination are counted. Lower the value of delay, better is the protocol. In this work, it is measured in millisecond.

$$\text{EED} = \sum(\text{arrive time} - \text{send time}) / \sum \text{number of connections} \quad (1.3)$$

d) **Consumed battery energy**: It is used to measure energy loss at a node. In this work, it is measured in Joules.

e) **Network size Vs Routing Load**: Routing load is the number of packets sent per data packets received to destination. However the packets are routing packets. It may very as per the network size.

f) **Network size Vs MAC Load**: This metric tells about the percentage of dropped packets at MAC layer.

g) **Network Size Vs Overhead**: Overhead is the addition of total generated and forwarded control and data packets.

1.5 Adhoc on-demand Routing Protocol

As per RFC 3561, Adhoc On-Demand Distance Vector (AODV) Routing Protocol [13] is distance vector routing protocol. In Distance Vector Routing, every node knows its neighbor (node within transmission range) and the cost (e.g. distance or number of hops etc. between two nodes) to reach them. Let s be the transmitting node and d is the destination node and x maintains the list of the neighbour node, then each node s maintains the set of distances D_{sx}^{d}. After regular time interval, every node broadcast this information in the form of the routing table, to all other nodes. Node s selects the node k as the intermediate node for which D_{sk}^{d} is minimum. This means every transmitting node selects the intermediate node for which the path length is minimum. Second most common characteristic of AODV is that AODV is reactive as opposed to proactive routing protocol i.e. AODV request a route when needed and does not require nodes to maintain the routes to destinations that are not actively used in communications. Due to its reactive nature, AODV incorporates less overhead and routes are determined whenever required and the unused routes are never determined. But it requires higher transmission delay because routes are determined before sending the packet. In AODV, nodes are identified with the help of IP addresses. AODV treats IP address just as a unique identifier. Routing table belonging to a node in AODV has following components:

- *Destination IP address*: IP address of a node for which data packet is destined.
- *Destination sequence number*: Every entry in the routing table of a node is assigned a destination sequence number which keeps on increasing with time. Every time in case more than one route options, destination sequence number with the highest value is chosen so as to find the latest path. Every node increments destination sequence number in two cases. If the node generates route discovery or route reply in a response to route request.
- *Next hop*: This value determines the next intermediate node.
- *Hop Count*: The total number of the intermediate node from source to destination.
- *Lifetime*: It is the time for which the path is alive.

The functioning of AODV can be understood by understanding the control messages that are sent from one node to another to determine the route. Control messages

are the messages that are sent from one node to another to implement the functioning of AODV.

1.5.3 Control Messages: Three main types of control messages are: Route Request (RREQ), Route Reply (RREP), and Route Error (RERR).

a) Format of Route Request (RREQ) Control Message: Whenever a new route is requested, RREQ is generated by the source node and forwarded by an intermediate node. Type field for RREQ message is 1. As shown in figure 1.5, Flag field determines the multicast or unicast nature of AODV. Whenever RREQs is generated, it is also assigned a broadcast ID. The broadcast ID of RREQ determines whether RREQ message is duplicate of previous one or new.

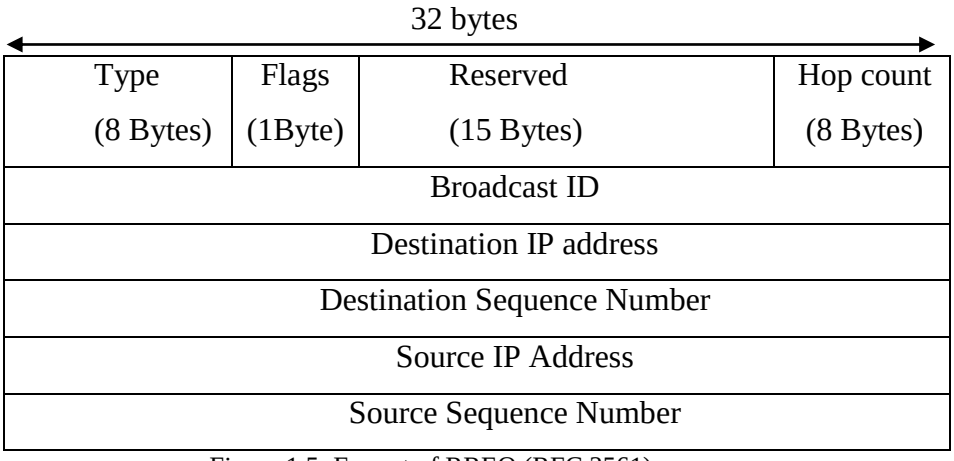

Figure 1.5: Format of RREQ (RFC 3561)

b) Format of Route Reply (RREP) Control Message: RREP messages are generated in response to the route request messages as shown in figure 1.6. RREP is type 2.

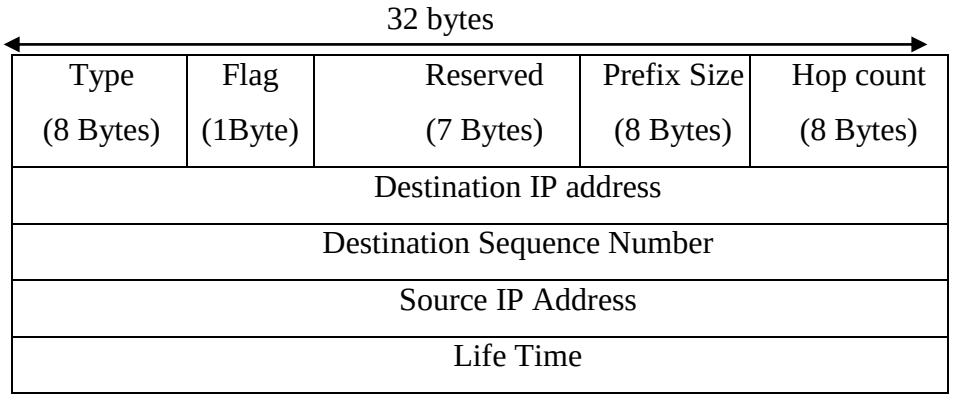

Figure 1.6: Format of RREP (RFC 3561)

Life time is the time for the route is valid and stored in the node receiving the RREP. It is usually measured in milliseconds. Prefix size denotes next hop that used for any node.

c) Format of Route Error (RERR) Control Message: As shown in figure 1.7, there must be at least one unreachable destination address. Destination count is the count of total unreachable destination.

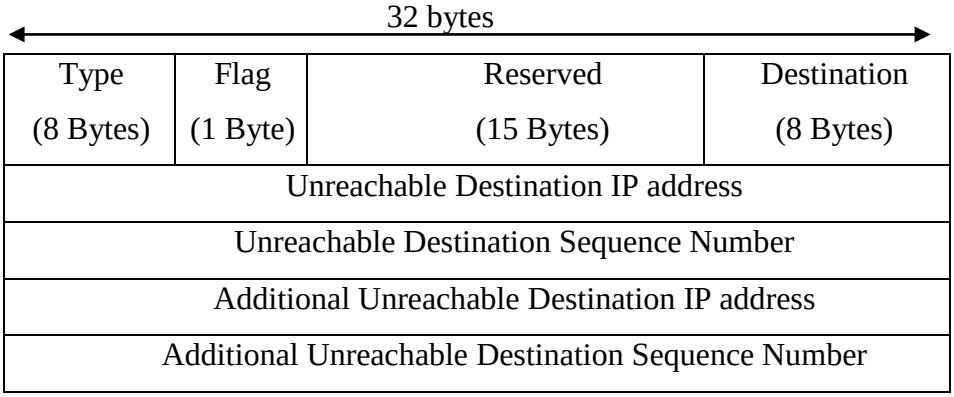

Figure 1.7: Format of RERR (RFC 3561)

1.5.4 Phases of AODV protocol

Main phases of AODV routing protocol are: Route Discovery and Route Maintenance [4].

a) Route Discovery: Every node keeps track of neighbor node by broadcasting the HELLO message at the regular time. Every node maintains the sequence number of the IP address of the destination node. This information is called 'Destination Sequence number'. This information is updated based on the information of RREQ, RREP or RERR. The sequence number is incremented if node originates route discovery process or destination node originates the route reply process. As shown in figure 1.8, whenever source node wants to send the message to the destination node, it broadcast a Route Request message (RREQ) to its neighbor, sets the value of broadcast ID and source IP address and waits for route reply (RREP). If the reply is not within limits then RREQ is broadcast again with incremented broadcast ID.

After sending the RREQ message, the intermediate node first increments the values of hop count. Then it further checks if it is the destination node or not. If it is the destination node then its sequence number is checked for path freshness (latest path with optimal values) otherwise RREQ is rebroadcast with its own IP

address till the destination is found. A node generates the RREP message if it is the destination or if it has an active route to the destination as shown in figure 1.9. RREP message is unicast back to the source node as shown in figure 1.10. When a reverse route is created, the following actions are carries out i) destination sequence numbers are compared ii) destination sequence number field is set iii) fields of next hop and hop count is set.

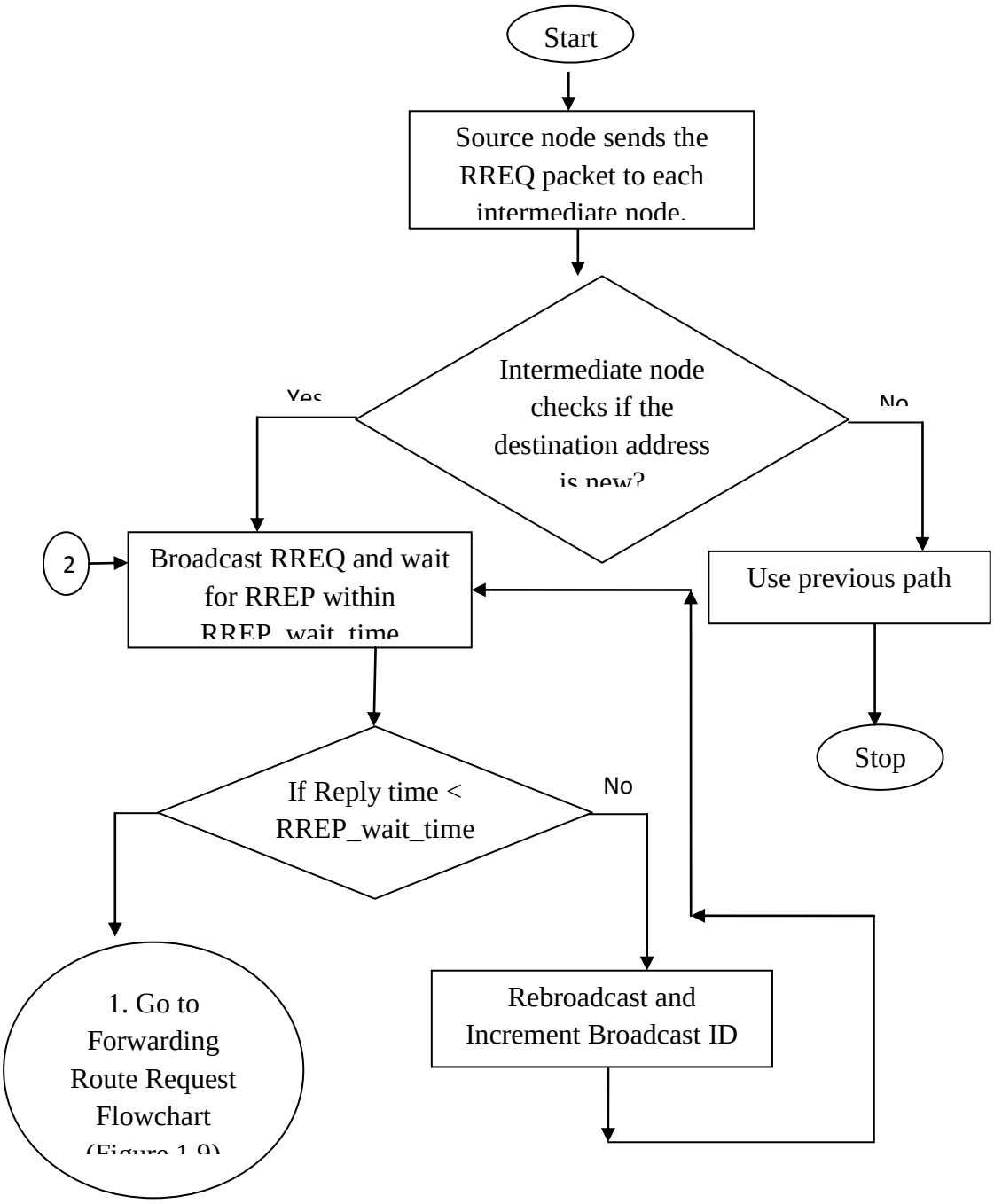

Figure 1.8: Flowchart the showing generation of RREQ control packet

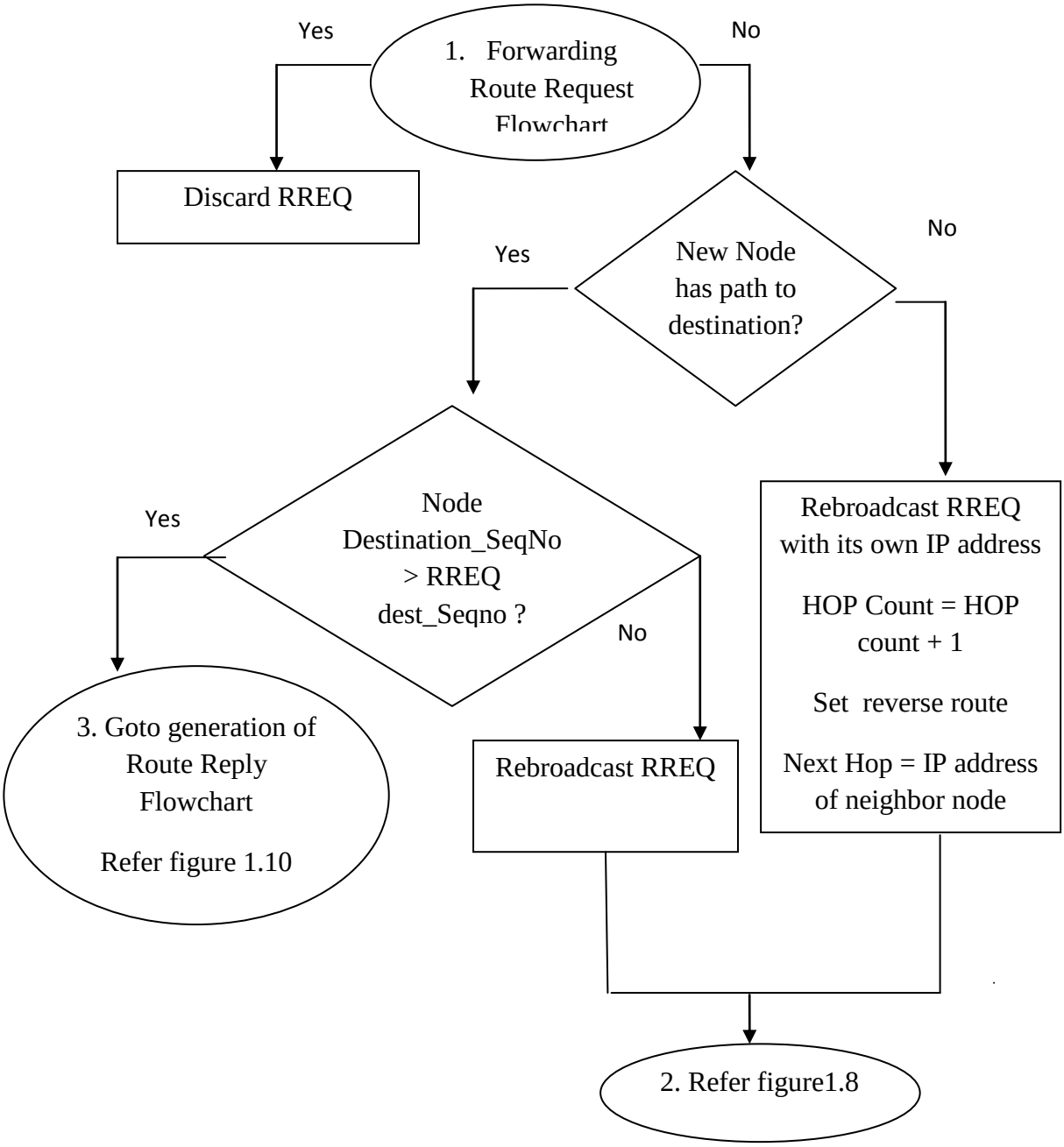

Figure 1.9: Flowchart showing the forwarding of RREQ

b) Route Maintenance: Maintenance of routes is done with the help of hello messages (a special RREP with hop count =0). If the route is active then neighbor node keeps on broadcasting the HELLO message to each other. If HELLO message stops coming then the neighbor node can assume that the other node has turned down or the link is broken.

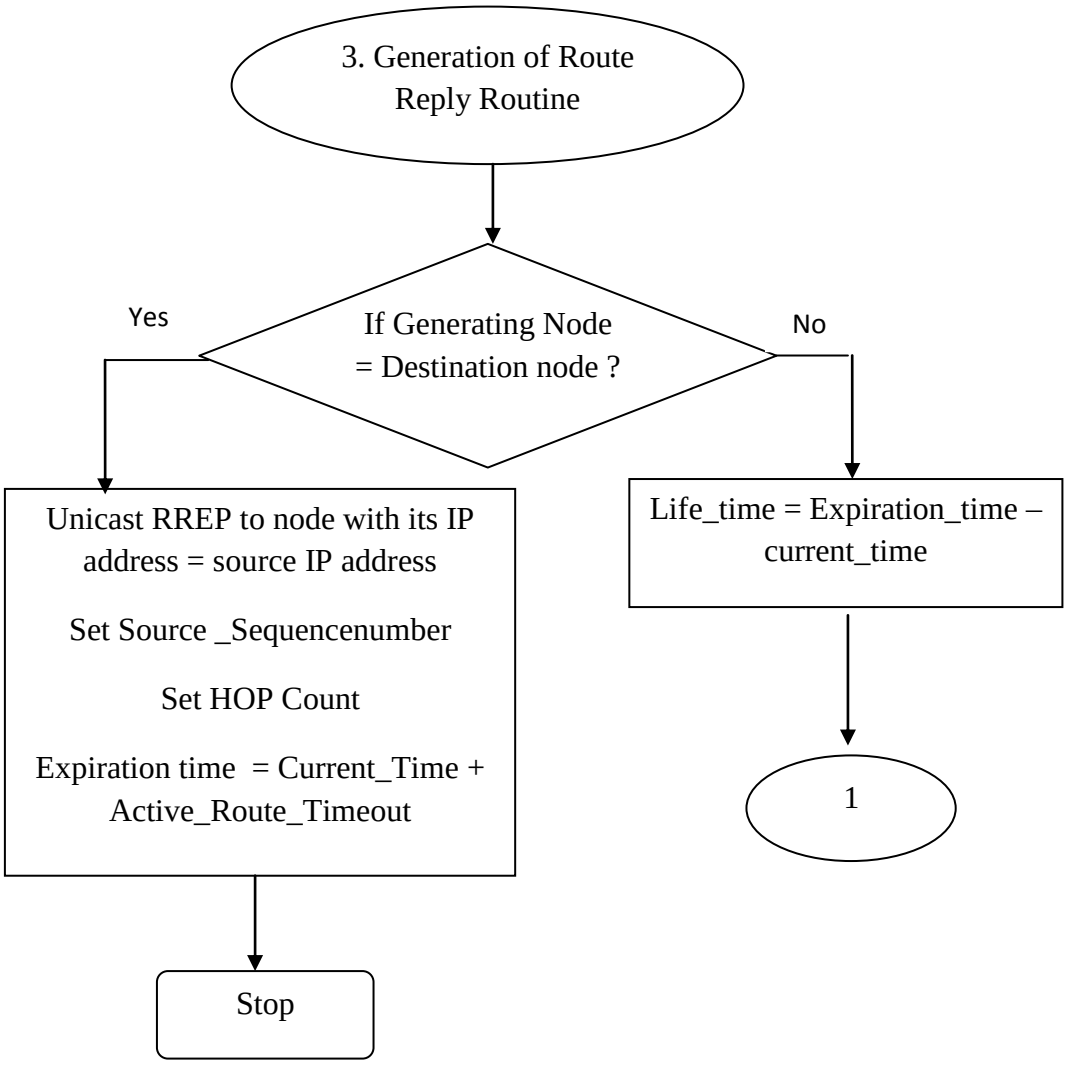

Figure 1.10: Flowchart showing generation of route reply

For example, Suppose S is the source and D is the destination node in the network of five nodes as shown in figure 1.11 (a). Initially, S transmits RREQ to its immediate neighbor of A and B. After receiving the route request message, the intermediate node A and B checks to see that whether it has a path to the destination. In the following network, A has a direct path to destination D while B's next neighbor is C. Therefore it chooses the route through A to D because the path through B is longer than A. At last as shown in figure 1.11 (e), D establishes a reverse route. D drops duplicate RREQ and A establishes a route. A unicast RREP and at last S establishes the route and all the unused routes are expired.

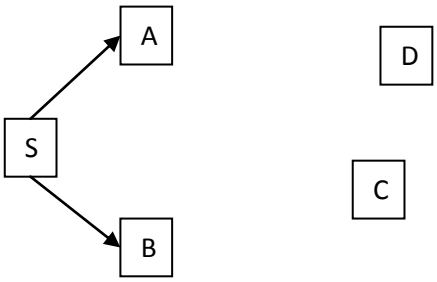

Figure (a): S sends Route request to A and B

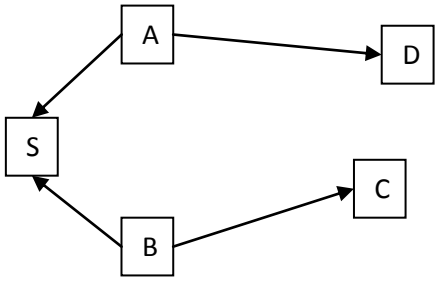

Figure (b): A and B sends RREQ to next neighbor

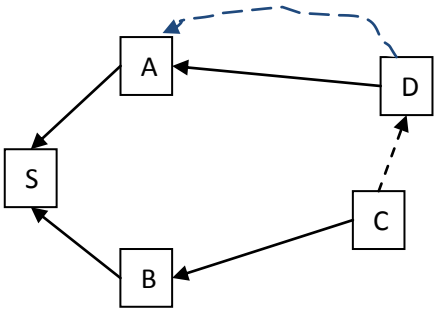

Figure (c): D generates route reply

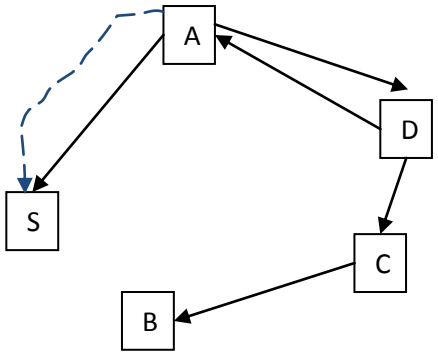

Figure (d): C Rebroadcast RREQ and D Unicast RREP

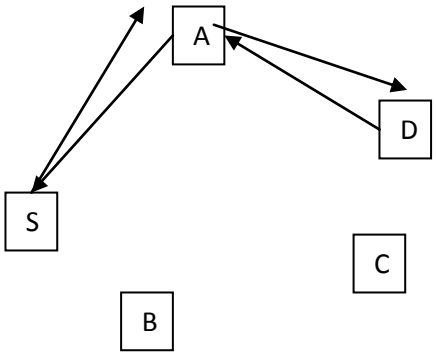

Figure (e): Path establishes between C and D

Figure 1.11: Routing in AODV, S is the source node and D is the destination node.

1.6 Research Gaps

The Following gaps have been identified from literature Survey

a. The existing security proposals in MANETs are attack oriented in the way that they first identify the attack and enhance the existing security protocol or propose a new protocol. However nowadays, several security architectures have been proposed that handles a number of existing or future attacks at the same time [41-44].

b. It is difficult to design secure routing protocol not only because of the characteristic of MANETs, but there is enough scope to improve upon formal analysis [45]. Formal analysis, on the other hand, provides the systematic way of discovering the flaw. Because the absence of formal verification can lead to security errors remaining undetected. Earlier M. Burrows introduces BAN logic to logically analyze a secure routing protocol but it too has some weaknesses. Further, Rubin Logic can be used to specify and analyze various protocols that use symmetric and asymmetric key under homogeneous and non-homogeneous environment.

c. Existing protocols can be made more secure after correct identification of selfish node [46,47] but identification of selfish node also varies according to change in topology or according to mobility environment. More work can be done on the identification of the selfish node in a heterogeneous environment.

d. Trust based scheme [48] can be integrated with security protocol [49] so as to check cooperation among selfish nodes. The main problem in today's network is that nodes do not in routing or make false recommendations. All those selfish or malicious behaviour can be understood and avoided by integrating trust metric to the node.

1.7 Problem formulation

Based on the finding of literature available and objectives, the following problem has been formulated for research work:

a. Security Architecture of Routing Protocol – The Trust architecture can be designed that can also handle different types of unpredictable attacks. Three different ways in which a node can engage itself in different malicious acts are:

- A selfish node may not or selectively forward the packet in the network.
- A node may not forward the packet to isolate the node from the other node.

- A node may advertise false route error message. In this false route request messages are injected into the network. This leads to wastage of bandwidth and thereby disrupting network operations.

b. Evaluation and Analysis of Existing Secured Routing Protocol – Based upon defined parameters existing secure routing protocol can be analyzed by following methods, which can help us in the understanding of weaknesses of security protocol.

- A Routing protocol can be precisely defined using some Formal method, which will help to prove various security flaws.
- Analysis of secured routing protocol can be carried out by simulation, using synthetically generated data sets. This would help in better understanding of performance and behaviour of routing protocol under different attacks.

c. Design and Development of the proposed Routing Protocol – After understanding the vulnerability of existing security schemes, the new protocol can be designed so as to add security to AODV routing protocol. New secured protocol will also intensify the efficiency in terms of throughput.

1.8 Research Objectives

The main objective of this research is to "Design and Development of an Efficient Secure AODV Routing Protocol". This aim is to be achieved through following steps.

a. To Analyze and Explore AODV protocol in order to find vulnerability against active and passive attacks.

b. To Design and Develop efficient secure Routing Protocol.

c. To Verify and Validate the proposed Routing Protocol.

1.9 Methodology

This work concentrates on the case of military rescue system. Detection of malicious nodes is a challenging task in rescue system. Further, isolating such malicious nodes from communication is also a great challenge. The trust based on the packet forwarding behaviour of neighbor can be used for detecting misbehaviour as discussed in previous works in literature. This model has been previously presented in several works of literature. But, by the same trust based logic, some of the neighbors those who were silent and not actively participated in communications will get wrongly identified as

malicious. So, simple trust based models will mark a lot of non-malicious nodes as malicious nodes. This will initiate a lot of link failures. Here, in this algorithm to keep track of selfish node and to distinguish between the selfish node and sleeping node (a node may be inactive due to no need of transmission of the packet), nodes in network keeps on sending the trust metric at the periodic time interval. All the nodes also maintain the trust metric of the node which is updated at the regular time interval. The updating is done based on feedback given by the previous node and nodes along with the best path. The best path is calculated based on distance and trust metric. In proposed work, it will overcome that problem and reduce the possibility of such false marking of non-malicious nodes as malicious nodes by introducing a Periodic Trust Handshake mechanism as shown in figure 1.12.

1.10 Our Contributions

a) An extensive background study on MANET and security issues in MANET has been presented

b) An extensive literature survey on previous works related to different kinds of attacks has been presented.

c) A reliable model for simulation of different kinds of attacks on AODV under NS has been developed.

d) An extensive evaluation and analysis of the impact of the different attack on AODV routing protocol with different network density have been made and the results are discussed in detail.

e) A novel method for detecting and preventing black hole attack and selfish node attack in AODV routing protocol using periodic trust handshake based malicious behavior detection mechanism (PTH-AODV) has been presented.

f) A novel method for detecting and preventing black hole Attack and selfish node attack in AODV Routing Protocol using dynamic trust handshake based malicious behavior detection mechanism (DTH-AODV)has been presented.

g) Comparison of performance of periodic and dynamic trust handshake based malicious behavior detection mechanisms with selfish node attack and black hole attack has been made and extensively analyzed.

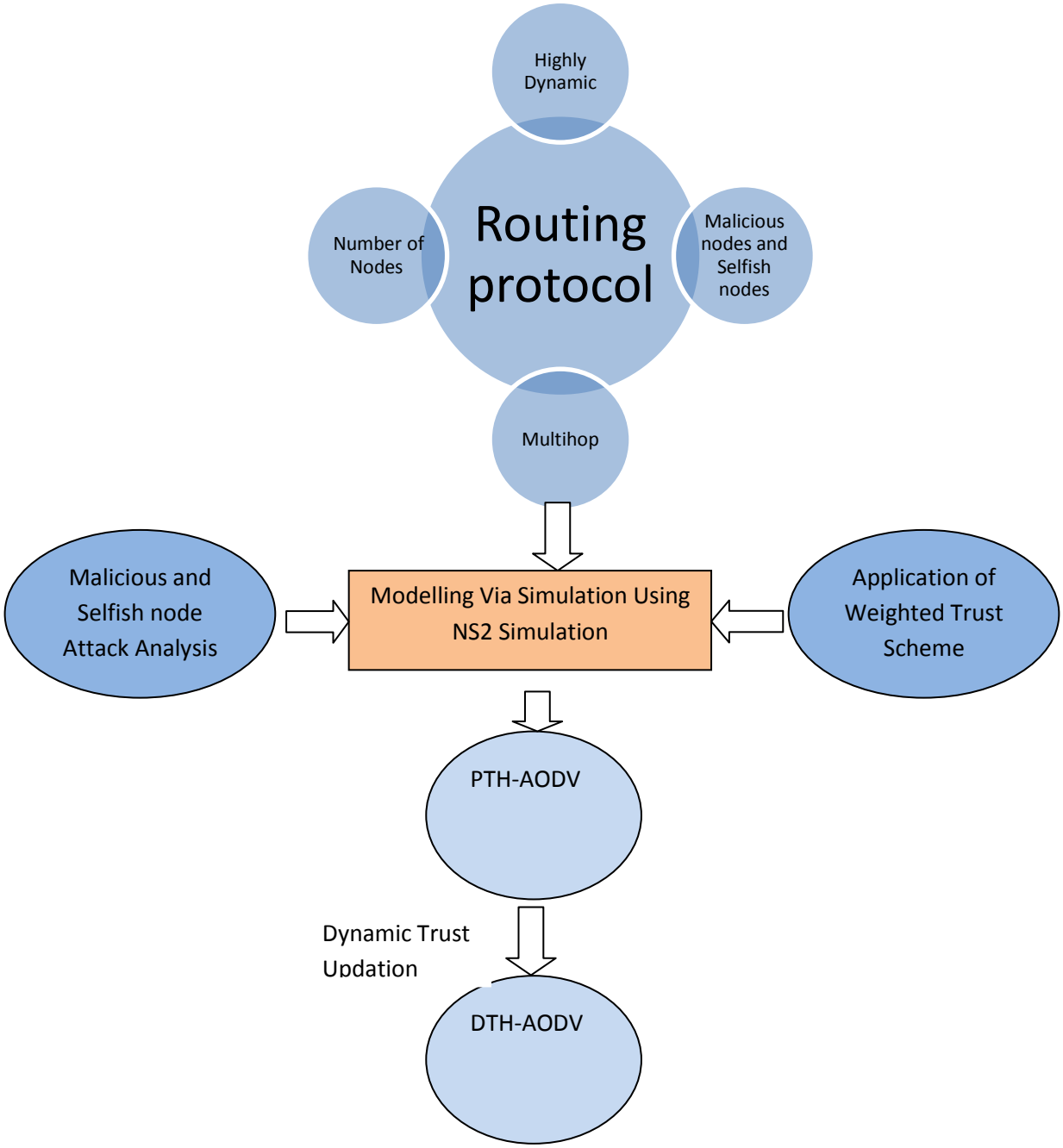

Figure 1.12: Modeling of proposed work

1.11 Thesis Organization

The thesis is divided into six chapters. Concise information regarding each chapter is as follows:

Chapter 1: It presents the introduction to MANETs and routing protocols. The routing protocols are categorized as i) proactive routing protocol ii) reactive routing protocol iii) hybrid routing protocol and iv) position aided routing protocol. Various categories of routing protocols are discussed in detail. Their emergence with time is also drawn in the figure. Further, it studies pros and cons of various routing protocol in terms of latency (time required to establish the network) and performance. This categorization helps us to understand the nature of routing protocols and identify the routing protocol that suits the requirement of short term military rescue operations. Adhoc on-demand distance vector routing protocol (AODV) which is the base protocol for the current work is discussed in detail. The performance metrics such as number of sent packets, receives packets, throughput, PDF, EED etc. are helpful to determine the performance of routing protocols etc. and hence are discussed in this chapter.

Chapter 2: This chapter discusses the vulnerabilities of AODV. The security is one of the major concerns now days. This chapter presents various attacks in MANETs and attacks in AODV in specific. To understand the working of AODV in presence of attack, nature of attacks and how various active and passive attacks the performance of protocol, this chapter chooses three common attacks namely i) Jellyfish attack (security attack classification 1: active attack (reordering of packet) ii) Selfish node attack (security attack classification 1: passive attack (no forwarding) iii) Black hole attack (security attack classification 1 : active attack (packet modification). The existing security enhancements in routing protocol is reviewed in detail and classified as three categories i) cryptographic based routing protocols ii) reputation based routing protocol ii) embedding trust metric into routing protocol. This chapter gives us a pavement to select and embed the weighted trust metric into routing protocol to increase the security of the protocol.

Chapter 3: This chapter gives the details of various analytical tools. The NS simulation tool is discussed in detail. It further presents the deep insight into attacks by pseudo code of attacks. It implements the three attacks in NS2. The impacts of three attacks on AODV routing protocol is compared and discussed to understand the nature and working of attacks.

Chapter 4: It gives details of our work. The title of our work is to propose a **"Design and Development of an Efficient Secure AODV Routing Protocol".** The Periodic Trust Handshake based trust AODV (PTH-AODV) proposed in this work will overcome the problem of identifying malicious and selfish node and reduce the possibility

of such false marking of non-malicious nodes as malicious nodes. A simple Periodic Trust Handshake mechanism will help to prevent such false identification.

Chapter 5: This chapter presents in detail dynamic packet forwarding based trust AODV (DTH-AODV). The main advantage of the proposed detection and prevention schemes is : they will detect and prevent the malicious nodes in the very early stage of AODV route discovery process. So, they will not need any manipulation in routing tables in the route resolving process, because, by the design, they will avoid including malicious hops in routing table even at the route discovery process itself.

Chapter 6: This is the concluding chapter, conclusions are drawn based on results obtained and future scope is outlined in this chapter.

Chapter-2

Literature Review

Due to vulnerabilities of AODV routing protocol, it is susceptible to the number of passive and active attack. In this chapter, the impact of various attacks on AODV performance is seen. The attacks to be chosen for study will play a major role in design and development of the new protocol. So keeping in view the security attack classification, attacks are chosen for study in this work, are selfish node attack and malicious attack especially black hole attack and jellyfish attack.

2.1 Vulnerabilities of AODV

AODV is chosen as the base protocol for the current research work. After discussing the working of AODV routing protocol, vulnerabilities of AODV are presented in this section such as [50]. Here malicious node is non genuine node in the network that works with the intention of harming the network.

a) A malicious node can drop any of the control packet or data packets.
b) A malicious node can modify any field of the control packet and can then forward the packet to its immediate neighbor.

The contents of this chapter have been peer reviewed and accepted for publication

[1] Bhawna Singla, A. K. Verma and L. R. Raheja, "Performance Analysis of AODV in Presence of Black hole Attack " Proceedings of the National Conference on Advancements in the Era of Multi Disciplinary Systems (AEMDS-2013) Elsevier Publications 2013 p.p 241-244 TERRI, Kurukshetra.

c) The malicious node can send the faked RREP or route reply acknowledgment (RREP_ACK) in response to the control message or it may send fake response message of its own.

d) In such way, the malicious node may cause the route breakage which may lead to node isolation or flooding of packets which may lead to resource consumption. Due to property that malicious node can also modify fields of the control packet, the malicious mode may impersonate any other node or it may leak the confidential information to the unauthorized node.

2.2 Classification of Attacks

The attack can be described as action taken against the target with the intention of doing harm. Attacks are targeted to damage basic aspects of security like integrity, confidentiality and privacy [52]. As in MANETs, every node is a router but some nodes perform it in a negative way. The nodes performing adverse effects on MANETs are classified into two categories: malicious node [53] and selfish node [54]. Malicious nodes are those nodes that perform an active attack on MANETs and may be active in route establishment or data forwarding phase, while selfish node performs passively by not forwarding the packet just for sake of saving battery energy. As shown in figure 2.1, attacks in MANETs can be divided into groups in a different way.

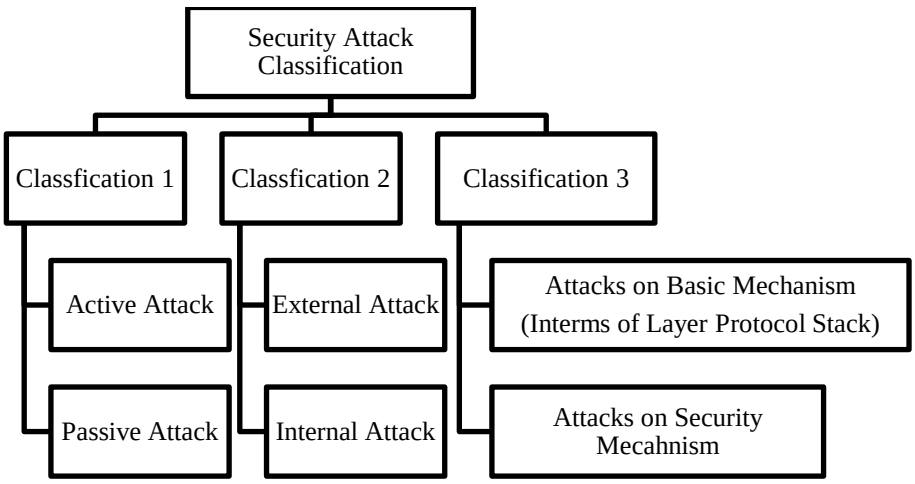

Figure 2.1: Attacks classification

a) **Security Attack Classification 1:** Attacks can also be classified as the passive attack and active attack [56, 57] as shown in table 2.1. The main goal of the passive attack is to monitor the working of the network without disrupting the operation of communication. Examples of passive attack include traffic analysis, sniffing to compromise keys etc. The main type of passive attack is the selfish node attack which aims at packet interception and energy consumption attack.

- Packet interception attack: In this attack, the selfish node does not forward the packet routed through itself to the destination.
- Energy consumption attack: Here, the selfish node, do not participate in the routing function in order to save energy.

Whereas Active attacks involve replication, modification and deletion of exchanged data. Some examples of active and passive attacks are shown in table1.1. The types of active attack are routing protocol poisoning, Routing table poisoning, Routing loop attack, Spoofing, Replaying etc.

- Routing protocol poisoning: In this attack, the attacker causes the problem in the working of the protocol so as to create congestion, partitioning of nodes, choosing higher routing path etc.
- Routing table poisoning: Here, the malicious node inserts incorrect information in the routing table. For example, the malicious node may claim itself as having the shortest path.
- Routing loop attack: Here, malicious node tries to send the packet in the loop instead to the destination.

Table 2.1: Security attack classification 1 [55]

Passive attack	Eavesdropping, Traffic Analysis, Monitoring etc
Active attack	Routing protocol poisoning, Routing table poisoning, Routing Loop attack, Spoofing, Replaying, etc.

- Spoofing: In this attack, malicious node claims to have different identity in terms of different IP address or different MAC address.

- Replaying: In replaying, the same messages keep on moving the network instead of transmitting to the destination.

b) **Security Attack Classification 2**: Attacks can also be classified into External and Internal attack [58]. External attacks are employed by the nodes that are not the part of network. They can be prevented by using standard security mechanisms such as firewalls or encryption. Internal attacks are carried by the nodes that ae the part of network but have become compromised node. Internal attacks are more severe attack since the malicious node belongs to the network as the authorized party and thus protected.

c) **Security Attack Classification 3**: Attacks can be classified as attacks on the basic mechanism of adhoc network and the attacks on the security mechanisms based on whether the attack is on the routing protocol or the underlying security technique. This chapter focuses on attacks on the basic mechanism.

Attacks on Basic Mechanism: Bing Wu et al. [59] classified the attacks on the basic mechanism according to layer protocol stack [59] as shown in figure 2.2. These attacks correspond to various layers such as physical layer, link layer, network layer, transport layer and the application layer. Attacks in network layer are further divided into the two phases: Route discovery phase and data forwarding phase. Figure 2.2 summarizes the attacks on these layers. This work mainly focuses on network layer attack. In MANETs, the main working of network layer can be divided into two phases: Route discovery phase and data forwarding phase.

- **Attacks at the Route Discovery Phase**: During route discovery phase, attacks may be due to fake routing updates. Attacks refer to any action of advertising routing updates that do not follow the specification of the routing protocol.
  - Routing message flooding attacks mean flooding messages such as Hello message flooding, Request message (RREQ) flooding, Acknowledgment (ACK) flooding.
  - Attacks using modification: Malicious node can cause redirection of network traffic and denial of service (Dos) [60] attacks by altering control messages field

or by forwarding routing messages with false values. DOS attack may further consume all the resources of the network leading to resource consumption attack. Examples of such attacks are redirection of traffic by modifying fields such as sequence number or hop count of control messages and Dos attack with modified source route.

- Attacks using fabrication: it may be due to sending of false routing message.

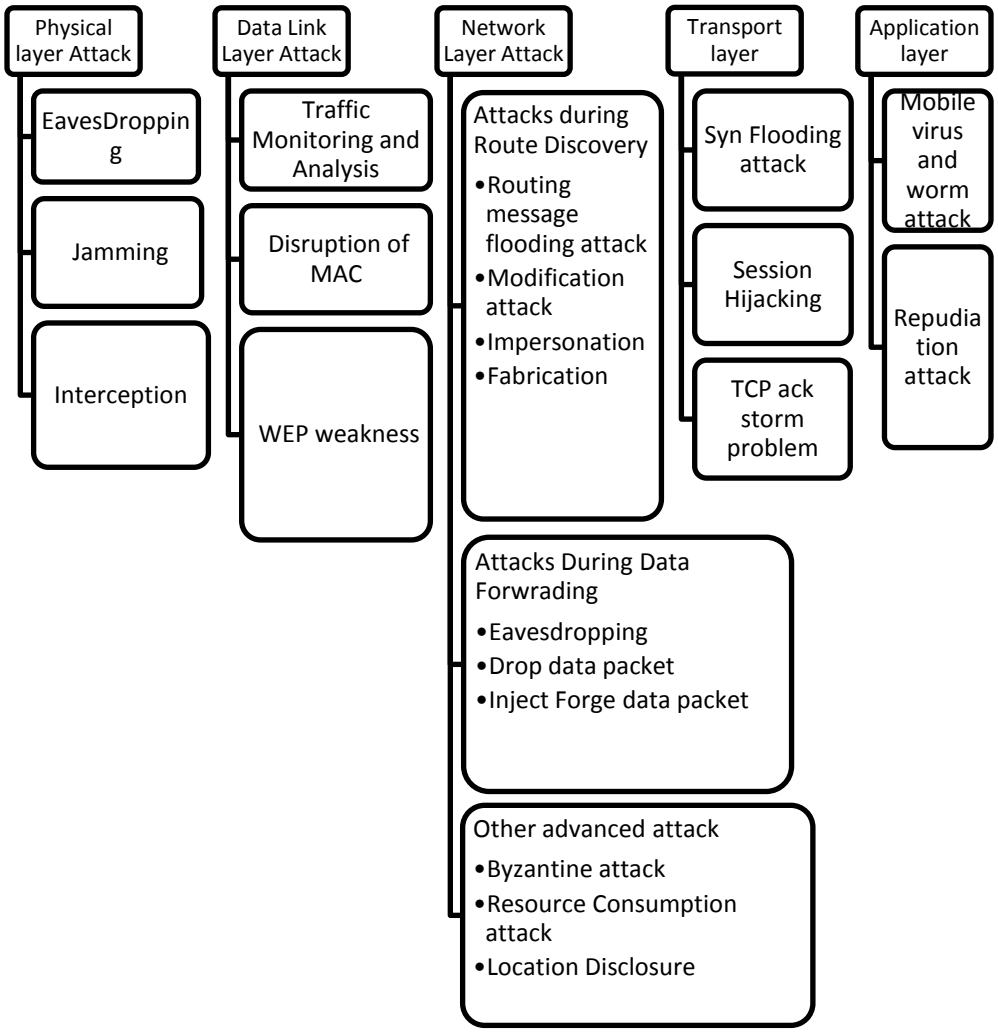

Figure 2.2: Layer wise description of attacks

❖ **Attacks at the Data Forwarding Phase**: Here node participates in route discovery and maintenance phases, but not in the data forwarding phase. Examples are

- Drop data packet: Malicious node can drop the packets it receives to forward.
- Fabrication: A malicious may fabricate data packet to overload the network.

2.3 Common Attacks in MANETs

a) **Gray Hole Attack** [61, 62]: In this attack, attacker initially behaves normally and participates in all kind of routing decision. But once it is chosen as the intermediate node, it drops the data packets.

b) **Black Hole Attack** [63]: this attack is similar to gray hole attack in the way that it misrepresents itself as having the shortest path. But after being selected as the intermediate node drops all packets (data as well as control) and drops the throughput. For example in the following figure 2.3, node 6 claims as having the shortest path, therefore, the destination node choose the path through node 6 instead of node 5 and node 3.

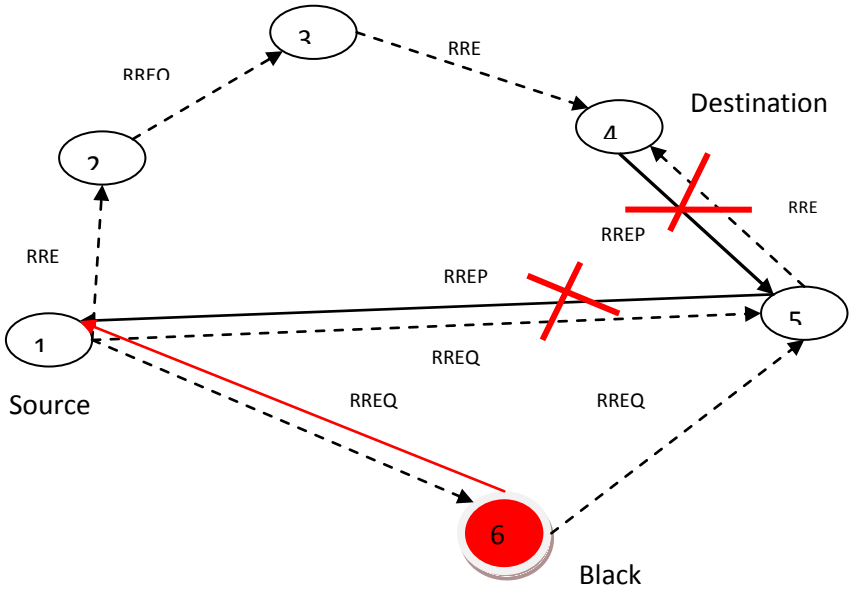

Figure 2.3: Black hole attack on the network where 1 is the source node, 4 is the destination node and 6 is the black hole node.

c) **Flooding Attack** [64]: In this type of attack, the attacker feeds the network by RREQ packet or the data packet. In case of RREQ packet, the malicious node tries to find the path to the non-existent node. Each intermediate node keeps on generating the RREQ packet to its neighbor node because it never finds the destination. Thereby it consumes all the resources available to it. In case of data packets, the malicious node injects the useless data packets into the network so as to create congestion.

d) **Wormhole Attack** [65,66]: In wormhole attack, adversary establishes a direct

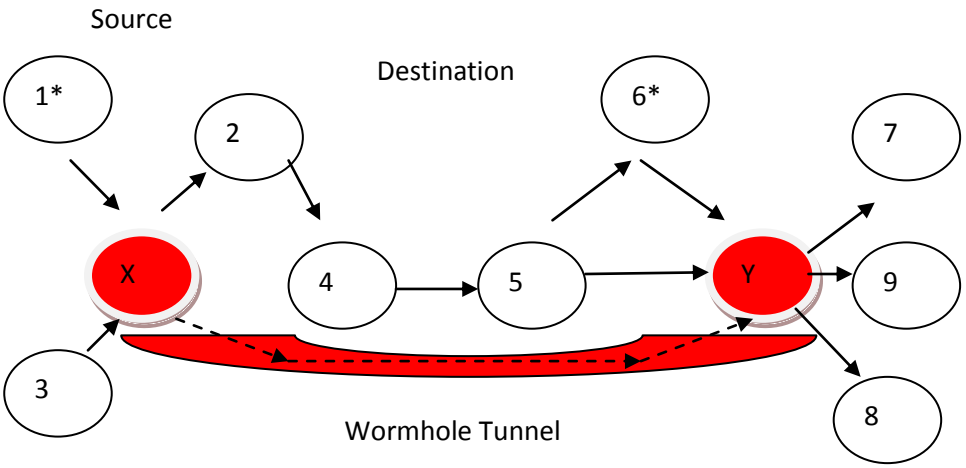

Figure 2.4: Wormhole attack in AODV routing protocol. Here, 1 is the source node and 7 is the destination node.

wireless link between two points in the network. The malicious node eavesdrops on the message at the one end of the link and tunnels them through the link to the other point in the network. Wormhole attack commonly involves two remote malicious node shown as X and Y where they are connected by wormhole link. Here, 1 is the source node and 7 is the destination node. During the path discovery process, node 1 broadcasts RREQ to a destination node 7. Node X being one of the intermediate nodes hears the RREQ. If the distance separation of the node X and 7 is larger than the communication range of the nodes than nodes will have to rely on the multiple hops paths to communicate with each other. In wormhole attack, an attacker uses the low latency link to rebroadcast the packets to destination node instead of using multi-hop path. A low latency link can be realized with a wired connection or wireless connection. As shown in figure 2.4, node X establishes a wormhole link with Y. When node 1 wants to transmit data to destination node 7, then malicious node X claims to have the shortest path. The malicious node X will forward the request through wormhole link to Y. At this point, node X and Y will establish a route via wormhole link, as if they were one hop neighbor.

e) **Jellyfish Attack** [67]: In this attack, the malicious node drops the throughput to nearly zero without disobeying the rules of the network. The malicious node does not perform any modification to the packet header. It just reorders the packet or delays the packet randomly. Due to this nature, it is very difficult to detect.

f) **Byzantine Attack** [68]: Here, the emphasis is on the survivability of network instead of the attack by one or more malicious node in the network. A weight is associated with each path from one node to another. When a node is found faulty, the weight is increased. When a source node wants to send the data from the source node to destination, the path with least weight is chosen.

g) **Rushing Attack** [69]: In this attack, the node receives the first RREQ message from the previous intermediate node. Node rejects the RREQ message it receives afterward. However, the first RREQ packet may denote the malicious intentions. After choosing the path with the malicious node, it may harm the network to any extent.

2.4 Attacks on AODV Routing Protocol

The impact of attacks [70-74] on AODV routing protocol can be better understood by understanding Transmission control protocol. Transmission control protocol (TCP) [75] is a transport layer protocol that is mainly responsible for i) ordered transmission of packets ii) retransmission of lost packets iii) congestion control. To ensure the delivery of packet when the source node sends a packet, the destination node sends the acknowledgment packet (ACK) back to the source node. Source node maintains the windows of packets for which is awaiting the acknowledgment. A timer is also set for the maximum time a node should wait for the ACK. If the time exceeds timer, it assumes the packet loss and retransmits the packet. TCP uses 3 way handshake protocols to establish the connection

a) SYN: whenever the server is ready for connection, the client sends a SYN to the server and sequence number is given a random value A.

b) SYN_ACK: In response to it the server replies by issuing SYN_ACK. The acknowledgment number is given a value which is one more then the value of A and sequence number of the packet is set to B.

c) ACK: The client sends ACK to the server. The sequence number is set to A+1 and acknowledgment number is set to B+1 thereby ensuring the connection.

Several extensions to TCP such as TCP Tahoe, TCP Reno, RED, TCP Vegas, New Reno, Sack TCP, and Compound TCP have been proposed.

Based on the vulnerability of AODV routing protocol as discussed in section 2.1 and attack nature, the most common types of attack identified in case of AODV are Jellyfish attack, Selfish node attack, Black hole attack. A detailed literature review of three attacks is presented in section 2.4.1.-2.4.3.

2.4.1 Jellyfish Attack

Jelly fish attack [76] is very prominent in the given scenario where the mobility is more and the route lifetime is short. Jellyfish attack by Aad et al. [76], is one of the denials of service attack. In case of distance vector routing protocol, the malicious node will obey all the control messages. However, once the route is established, it will reduce the throughput of the network via jellyfish attack. The goal of jellyfish attack is to reduce the performance of the network to near about zero without dropping zero or a negligible number of packets. Jellyfish attack can be classified into three categories [77-78]: a) JF reorder attack b) Periodic Dropping Attack c)Delay variance attack.

a) **JF Reorder Attack**: TCP has a vulnerability that the packets once transmitted sequentially may arrive at the destination in unordered sequence due to multipath ordering routing and route changes. No TCP variant is robust to malicious reordering of packets. Let ACK-N be the acknowledgment that all the segments from 1,…,N have been received. Then receipt of duplicate ACK-N will show the out of order packets.

b) **Periodic Dropping**: In this malicious node drops some percentage of packets for the maliciously chosen period.TCP throughput can become equal to nearly zero even for the small values of x where x is the percentage of the packet.

c) **Delay Variance Attack**: In this, the malicious node delays the packet while preserving the order in which packets are transmitted. Time to delay the packet is chosen randomly.

2.4.2 Selfish Node Attack

Whenever the selfish node feels that the packets require a lot of resources, the selfish node does not forward it in the network [80-83] as shown in figure 2.5.

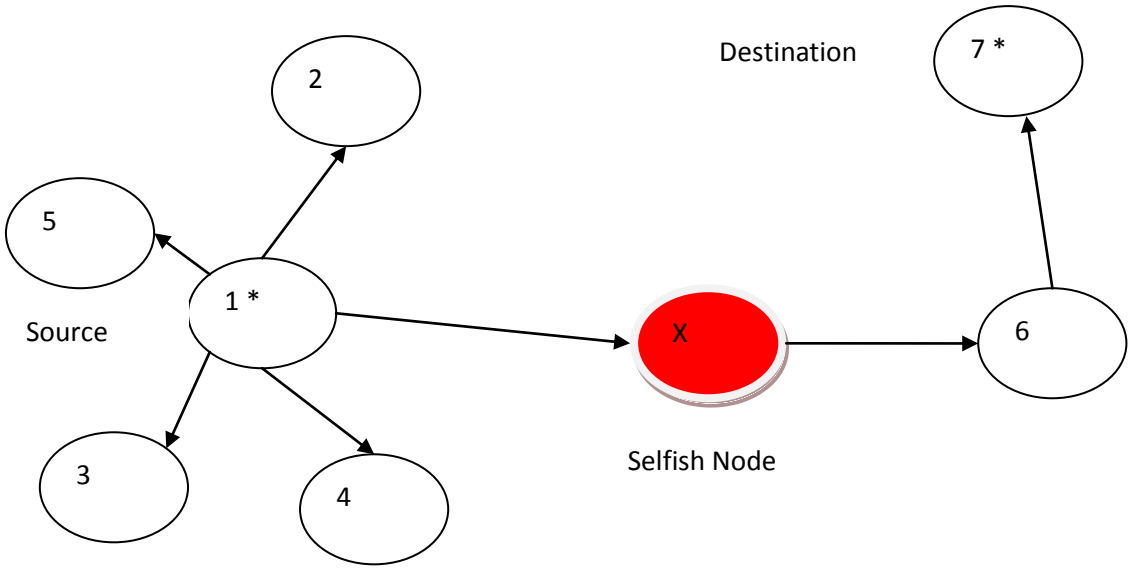

Figure 2.5: Network consisting of 7 nodes where 1 is the source node and 7 is the destination node and X is the selfish node.

nodes even if active nodes are still available. In figure 2.5, suppose node X is the selfish node. Suppose source node is node no. 1 and the destination node is 7, node 1 selects X as the next hop and sends data to it but X discards all the data forwarded to it or any other node at a distance of more than one hop away. In this way, node 1 will become isolated. However, selfish nodes can still make the communication with all other nodes (via cooperative neighbors).Selfish nodes are of three types:

a) **No Packet Forwarding:** Here, once the path is established the selfish node does not forward the packet.

b) **No Participation:** In this type, a selfish node does not participate in the route discovery phase of AODV protocol. Due to this network maintenance becomes more significant as compared to route discovery phase. If the node does not participate in the route discovery phase, then there is no route including selfish node, as a result, packet forwarding function will never execute.

c) **Partial Packet Forwarding with Energy Saving:** Here, node consumes more energy while performing all the network related operations.

2.4.3 Black hole Attack

Black hole problem is type of active attack in which malicious node first claims to have the shortest path. Source node chooses the route containing the malicious node to the destination. Once the traffic is routed through itself, it drops all the data packet routed through itself [84-87]. As shown in figure 2.6, let 1 be the source node and 3 be the destination node and 4 is the malicious node. 4 claims to have the shortest path that is why route through 4 (1-4-5-6-3) is selected instead of 1-2-3. But after being selected in the final route 4 drops the entire data packet. The working of black hole attack is further summarized in figure 2.7. The figure shows that if the packet forwarded is data packet and the node is malicious, then it drops the entire packet. Otherwise, if the packet if RREQ control packet and the node is malicious then it sends the fake RREP so as to claim itself as having the shortest path. Once it is chosen as the intermediate node, it drops the entire data packet routed through it. In all other cases, it behaves normally [88-90].

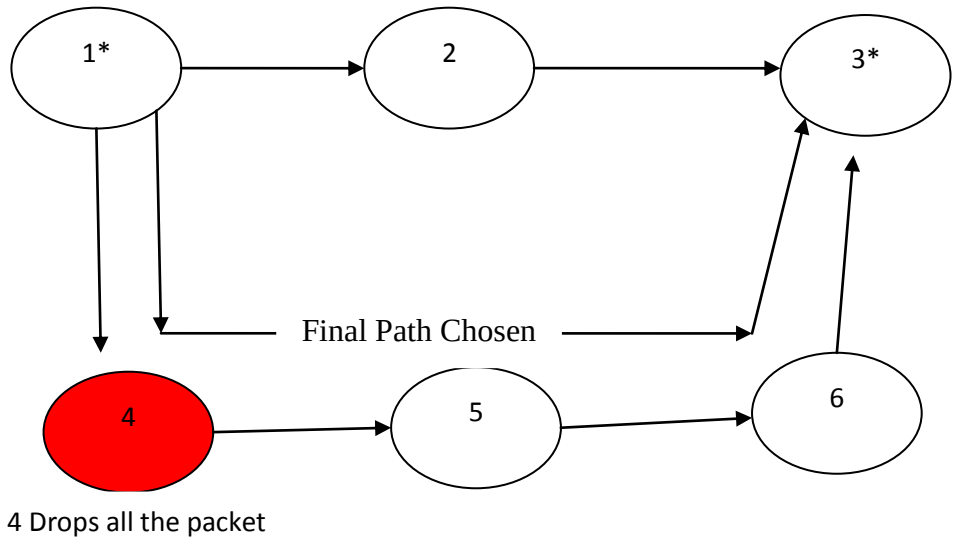

Figure 2.6: Example showing the working of black hole attack where 1 is the source node, 3 is the destination node and 4 is the malicious node.

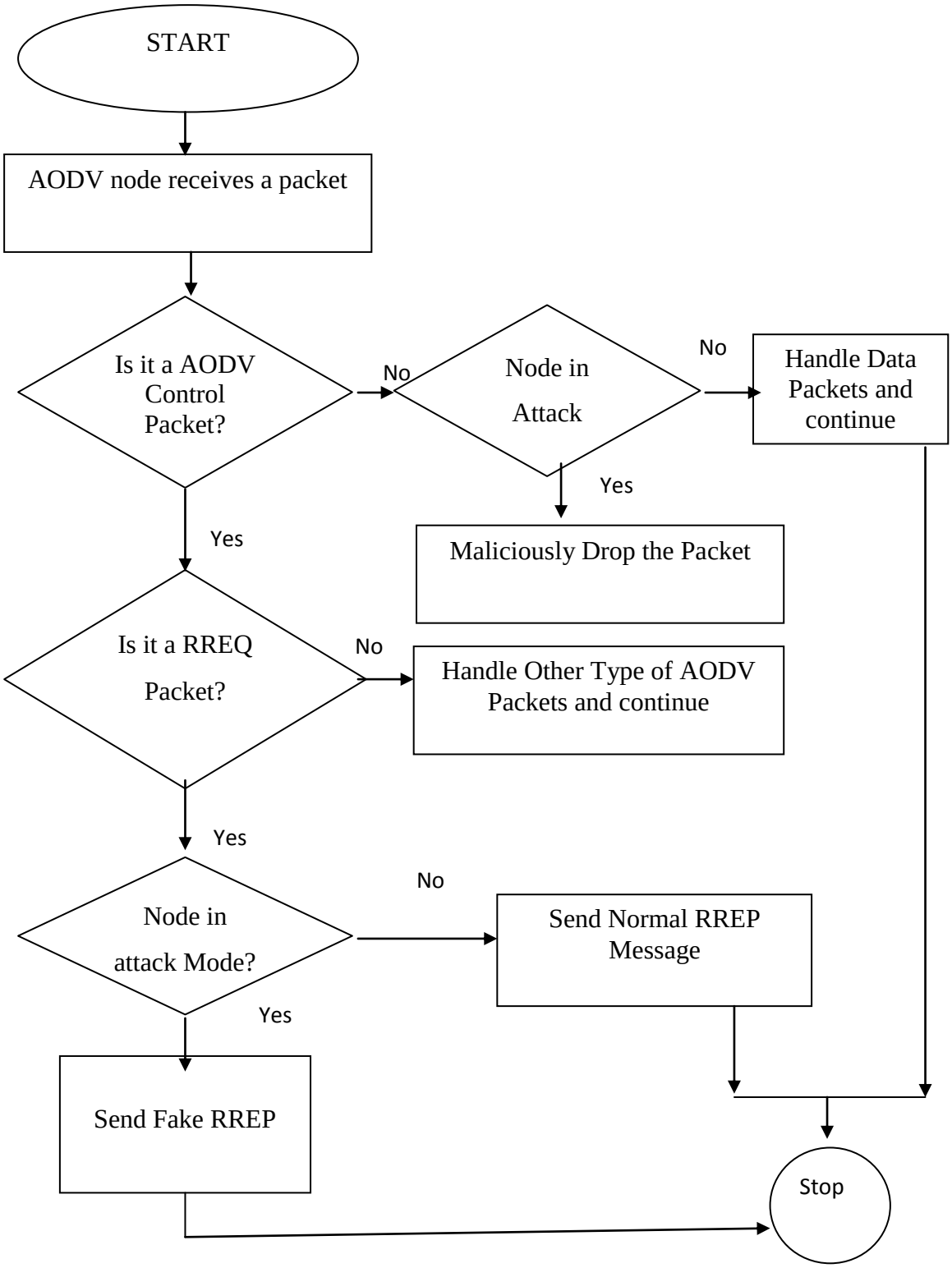

Figure 2.7: Flowchart showing the working of Black hole attack

2.5 Existing Security Enhancements to Routing Protocol

After studying the role of attacks on AODV routing protocol, section 2.3 presents the analysis of current security enhancements to AODV routing protocol. A lot of work [91-94] has been done to make AODV secure. To achieve the major security goals of confidentiality, integrity, authentication, non-repudiation, access control etc. and to avoid attacks, certain integrated security enhancements are added the routing protocol. These security measures are divided into three categories: Cryptographic based routing protocol, Reputation based routing protocol and Embedding trust metric into routing protocol.

2.5.1 Cryptographic Based Routing Protocol [95]

It is further divided into three types: Asymmetric cryptography solution, symmetric cryptography solution, hybrid solution

a) **Asymmetric Cryptography solution**: It uses public and private keys in each packet. Maintenance of keys is done by trusted third party server (TTP). The main function of TTP is to bind the digital certificate associated with public key with its identity. This category includes ARAN.

Authenticated Routing for Adhoc network (ARAN) [96]: It consists of three stages: (i) Preliminary certification, (ii) Route discovery and (iii) shortest path determination. In preliminary certification, each node contacts to TTP to obtain its address and public key. This protocol assumes that every node knows about its address and key before joining the network. The second stage, route discovery provides the end-to-end authentication. In this stage, sender node first transmits a route discovery packet (RDP) to its neighbor node that contains the certificate of initiating node, timestamp, nonce and the address of destination node. Each intermediate node after receiving the RDP, validates the signature and updates its routing table with the neighbor node and forward the packet to the next neighbor after removing the certificate and signature of the previous node. The destination node after receiving the RDP replies with the route reply (REP). The REP contains the address of the source node, the destination's certificate, a nonce, and the associated timestamp. The third stage is optional that determines the shortest path to the destination using shortest path confirmation packet (SPC).

b) Symmetric key Cryptography solution: It uses single symmetric key for cryptography and includes the following algorithms.

Secure Routing Protocol (SRP) [97]: Here, a shared secret key called security association (SA) is associated with the source and destination. The route request packet generated by the source node (query) contains query sequence (to identify the outdated request)(QSEQ), random query identifier (to identify the specific request)(QID) and output of key hash function. All the nodes also maintain the priority ranking of the neighbors. When a query reaches to the destination, it first checks it is not outdated or replayed by checking the QSEQ. Then it checks the integrity and authenticity by computing the value of the hash function. In response to the query, destination node generates a number of replies as many as intermediate nodes. The reply contains QSEQ, QID and message authentication code (MAC).

Secure Efficient ADHOC on-demand Distance Vector Routing Protocol (SEAD) [98]: This protocol use hash function to authenticate hop count and sequence numbers. Source node transmits a route request packet containing source and destination address, identification of RREQ, hash authenticator of sequence numbers and hop count. After receiving the RREQ packet destination node transmits the route. Each intermediate node that receives the route reply checks the authenticity and replaces the hash chain till the packet reaches to the sender.

Ariadne [99]: It is Adhoc on-demand routing protocol based on DSR developed by the author of SEAD. While SEAD employs hop by hop security mechanism, Ariadne employs end-to-end security mechanism. Ariadne uses shared secret key between each node and MAC to authenticate point to point message between these nodes. Ariadne employs the TESLA broadcast authentication protocol to authenticate broadcast messages, such as route requests. In TESLA a sender generates a one-way key chain and defines a schedule according to which it discloses the keys of the chain in reverse order from generation. Here, route request contains source and destination address, ID, TESLA time interval and two empty lists, namely a node list and a MAC list. Neighbor node after checking the validity of packet inserts its own address, replaces the hash chain and appends a MAC with MAC list till the packet reaches to the destination node. Destination node then generates the route reply packet.

c) **Hybrid solution**: In this secure routing protocol employs both the symmetric as well as asymmetric routing protocol.

Secure Adhoc On-demand Distance Vector Routing Protocol (SAODV) [100]: It uses digital signature and hash chain in order to utilize AODV. Here, digital signatures are used to authenticate non-mutable fields of RREQ and hash functions are used to secure hop-count field. When a sender node transmits RREQ, the RREQ packet is encrypted using hash function which is applied on the max hop count field and random number. An intermediate node that receives a route request or a route reply checks the integrity by verifying the digital signature. The hop count field is verified at the intermediate node. Before the packet is re-broadcasted by the intermediate node the value of the hash field is recalculated by the intermediate node.

Secure Link State Routing Protocol (SLSP) [101]: It provides the security in terms of secure proactive link state information. All the nodes broadcast their public key certificate within their zone using signed public key distribution packet. The link state information is also broadcasted internally using neighbor lookup protocol. All the nodes carry IP as well as MAC addresses.

Security Aware Adhoc Routing (SAR) [102]: Traditional security protocol uses distance, hop count etc. as a cost metric for routing operations but SAR uses security metric such as a trust for routing operations. The sender along with sending the route request message also sends the security requirement of the route. The neighbor node that responds to the packet also compares itself that whether it can fulfill the given security requirement. If it can fulfill then that node is chosen otherwise other node is searched.

Techniques for Intrusion Resistant Adhoc Routing Algorithm (TIARA) [103]: It is based on flow based route access control list. It means that each node maintains the list of authorized flow. If the route requested belongs to the list then it is used otherwise rejected. It also uses multipath routing. Moreover, the flow is also monitored by the destination node.

Building Secure Routing out of an Incomplete Set of Security Associations (BISS) [103]: All the route request messages are signed y the public key of the sender. If the intermediate node knows the security association with the sender then it appends the

message with the keyed hash. Otherwise, if t does not know the secret then it signs with its own public key and authenticates the node.

Packet leashes [105]: It is only used to avoid wormhole attack. Here the stress is given to identify that whether the packet has traversed unrealistic distance or not.

SANE-DSR [106]: uses genetic algorithms for path optimizations. It is pseudo DNA based cryptographic algorithm. Table 2.1 further presents the summary of cryptographic routing protocols.

Table 2.2: Summary of cryptographic routing protocol [107]

Routing Protocol	**Routing Approach**	**Requirement**
ARAN	Reactive	Trusted Third Party Server
SAR	Reactive	Shared Secret Key Mechanism
SEAD	Proactive	Existence of shared secret key
Ariadne	Reactive	Clock synchronization and shared secret key between each pair of node
SAODV	Reactive	Online key management
SLSP	Proactive	Uses public keys of node.
SAR	Reactive	Uses Security Associations
TIARA	Reactive	Public key infrastructure
BISS	Reactive	Shared Secret Association
Packet Leashes	Reactive	Clock Synchronizations
SANE-DSR	Reactive	Genetic Algorithms

Table 2.2 briefly describes all the commonly used cryptographic routing protocols. It also describes the main requirement of the protocol such as requirement of trusted third party server etc. Table 2.3 further summarizes the behavior of defense of

cryptographic routing protocol against certain common attacks like black hole attack, worm hole attack, selfish node attack etc. as suggested by Patroklos G. Argyroudis et al. in “Secure Routing for Mobile Adhoc Networks” [108]. It is seen cryptographic routing protocols are vulnerable to attacks such as black hole, selfish node and jellyfish attack.

Table 2.3: Defense against attacks [108]

Attacks/ Routing Protocols	**ARAN**	**SAR**	**SEAD**	**Ariadne**	**SAODV**	**SLSP**	**SAR**	**TIARA**	**BISS**	**PL**
Location disclosure	No	No	No	No	No	No	No	No	No	No
Black hole	No	No	No	No	No	No	No	Yes	No	No
Replay	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	No
Wormhole	No	No	No	No	No	No	No	No	No	Yes
Selfish node	No	No	Yes	Yes	No	No	No	No	No	No
Jelly fish Attack	No	No	No	No	No	No	No	No	No	No
Routing table poisoning	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes

Figure 2.8 (a,b) gives the timeline of cryptographic protocols developed in history (from 1999) till now.

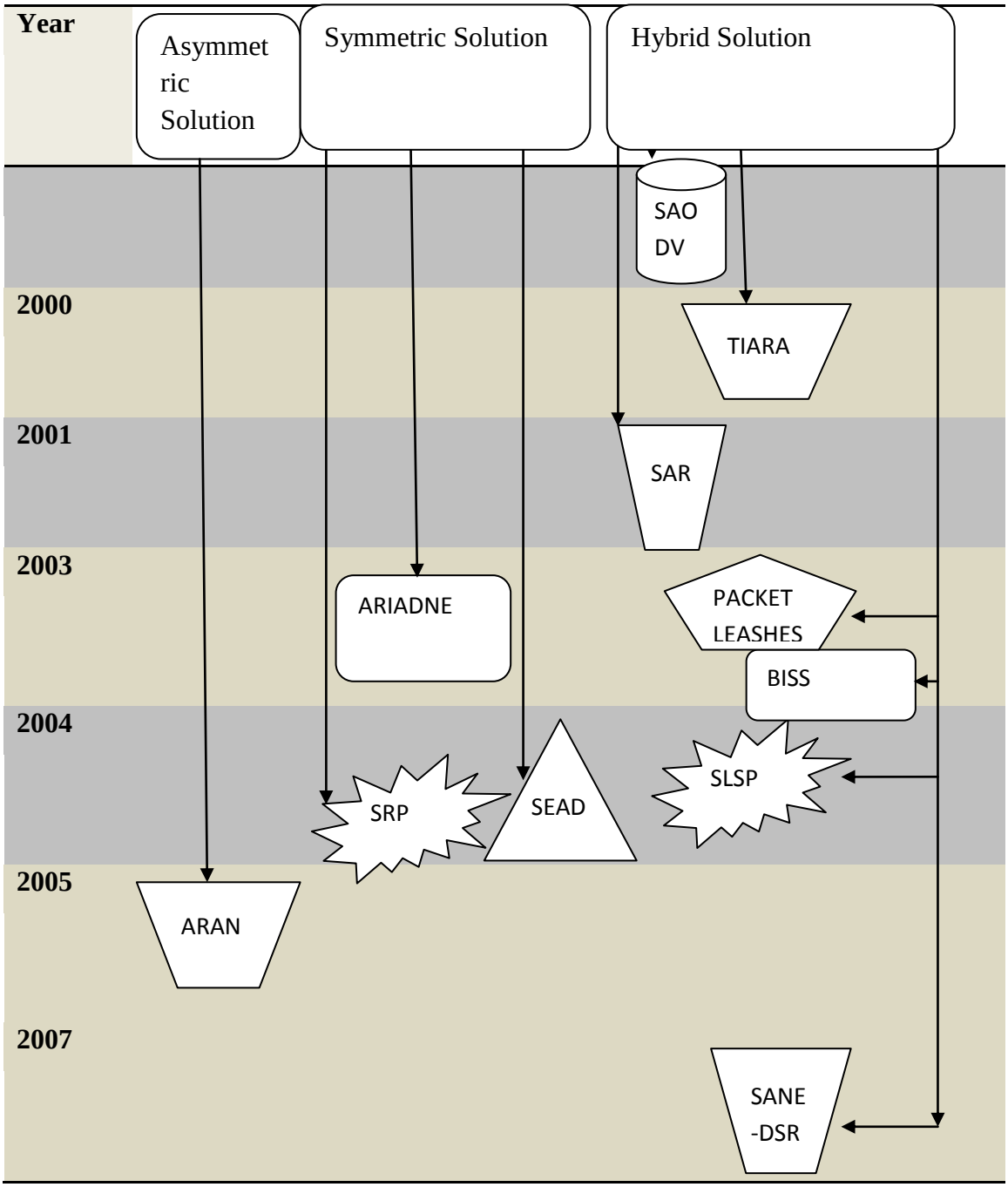

Figure 2.8 (a): More Common Cryptographic Solution of adding security in routing protocols (1999-2007). The AODV extensions are enclosed in ⛁ , DSR extensions are enclosed in ▽ , AODV / DSR extensions are enclosed in▢, DSDV extension is enclosed in △, hierarchical are enclosed in✸ .

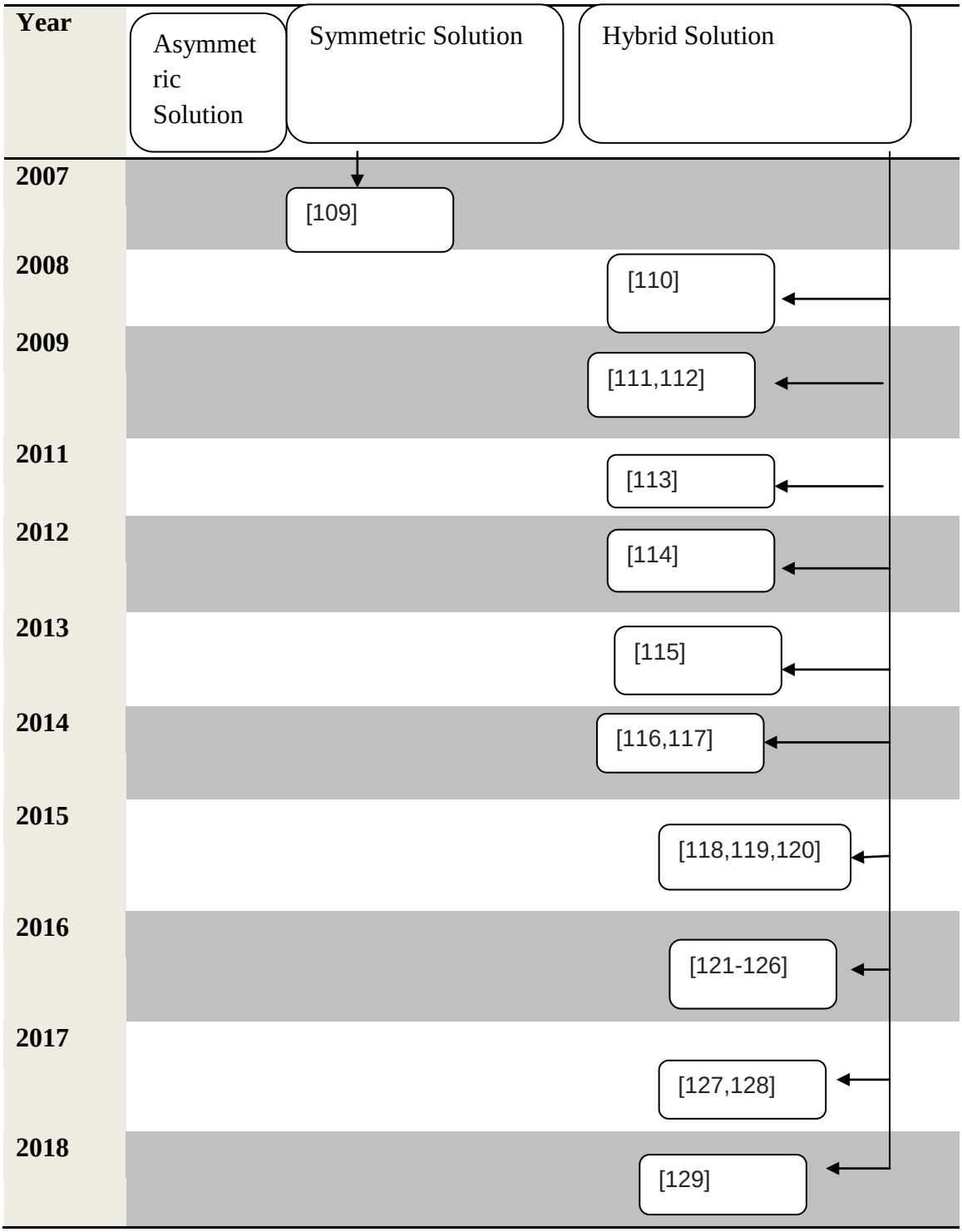

Figure 2.8 (b): Recent Cryptographic Solution of adding security in routing protocols (2008-2018).

2.5.2 Reputation based Routing Protocol [130-132]

Reputation metric of node helps to identify the extent to which that node can be trusted for its genuine participation. More a node has the reputation metric; more is a chance that a node is not an attacker node. Reputation metric is directly proportional to

cooperative forwarding by the node to next neighbor. They intend to reduce the selfish behavior of node thereby reducing the selfish node attack. Cooperation among the nodes can be increased by the following three methods (a) token based routing (b) payment based routing where cooperating node is given remuneration (c) reputation based routing where the cooperating node is pampered by assigning more reputation [133-136].

a. **Token based routing** [135]: Here each node is assigned a token with the lifetime in advance of communication. Token is the entity that defines the node cooperation in the network. If the node is benign then it need not renew the token otherwise token expires frequently. Only the node with the token can participate in the communication. Whenever a node is transmitting or receiving the data, its neighbor node observes its communication collaboratively.

b. **Payment based routing** [137,138]: In this routing, each node maintains a wallet where e-money can be kept. Whenever a node wants to deliver a packet to another node then source node pays its payment to the receiving node. In this scheme also neighbor nodes keep watch on the behavior of node. Misbehaving node is taken out from the network. The node wallet is protected from attacks by cryptographic measures. The most common example is Nuglet scheme.

c. **Reputation based routing**: Here the reputation is calculated based on node observation, neighbors observation and friends observation.

Kevin hoffman et al. in "A Survey of Attack and Defense Techniques for Reputation Systems" [139] suggested three fundamental phases of reputation based routing:

a) **Formulation**: In this phase, various metrics such as direct observation, indirect observation, feedback etc. of reputation factor is suggested.

b) **Calculation**: Here, reputation of each node is calculated under the given set of constrained.

c) **Dissemination**: In this phase, the reputation metric is shared with the rest of nodes.

Some examples include Watchdog and pathrater, CONFIDANT, CORE etc [157,158].

Watchdog and Pathrater [140]: In this scheme, each node watches its neighbor node for correct transmission of data. Node watches its neighbor node by listening to the network in the promiscuous mode. The listening node also checks that its neighbor node does not make any modification to the packet. If the node is not able to forward the packet within limited time period then its failure ratio is incremented otherwise success ratio is incremented. After determining the success metric of each node, the pathrater extension determines the path in which all nodes have maximum success metric.

CONFIDANT [141]: It identifies the non cooperating node based on observations of neighbors and isolates it from participating in the routing decision. Similar to watchdog mechanism, each node listens to its neighbor node. Once it determines that it is not forwarding genuinely then it generates the alarm message. The alarm message is exchanged by the neighbor node and friend node. Once it determines that the rating of a node is below the threshold level the reputation system marks its entry its table and the path manager delete that node from the path.

Cooperation Enforcement Based on Reputation CORE [142]: Here, final reputation metric combines direct as well indirect observation as well as functional reputation based on various functions. Each node maintains a reputation table where the reputation of a node increases if it forwards the packet otherwise decreases. It assigns more weight to old observations then the current observations and only positive reputation are exchanged between the nodes.

Table 2.4: Comparison of existing and common reputation based schemes.

Reputation Systems	Characteristics	Advantages	Disadvantages
Watchdog and Pathrater	If the node forwards the packet within threshold time then it is considered to be cooperative and its reputation is incremented	Able to detect selfish node to some extent.	Radio Propagation errors and packet collisions impacts its potential to detect selfish node

	otherwise decremented.		
Confidant	Node consists of four major parts: ➢ Monitors ➢ Reputation record for calculating first hand and second hand trust ➢ Trust record ➢ Path manager	Uses second hand observations to better detect the selfish node.	If a trusted node makes wrong accusation or a sufficient number of node make wrong accusation then confidant fails.
CORE	Each node maintains a reputation table which increase with successful forwarding and decided on direct as well as indirect observation	Gives more weighage to old observations avoiding temporary selfish behavior False accusations are avoided	If there is no interaction among certain number of nodes, it will deteriorate the performance of protocol.

2.5.3 Embedding Trust Metric into Routing Protocol

According to Eschenauer et al. [143], trust is defined as "A set of relations among entities that participate in a protocol." These relations are based on previous interactions among the nodes. The trust increases if the previous interactions are faithful. In another survey by Jin-Hee Cho et al. in "A survey of Trust management in Mobile Adhoc Networks" [144] trust should have following characteristics as shown in figure 2.9.

a) Trust should be dynamic. Because the nodes are highly mobile therefore information regarding trust can change rapidly. This is also a reason that trust is expressed in continuous-valued variable.

b) Trust is subjective. Two nodes may have different levels of trust in different time depending on different situations.

c) Trust is not necessarily transitive. If A trust B and B trust C it does not means that A trust C.

d) Trust is asymmetric. It means that a node with higher capability may not trust a node with lower capability. However, a node with lower capability may trust a node with higher capability.

e) Trust is context dependent. It means that if A trust B as an employee, he may not trust B as a broker.

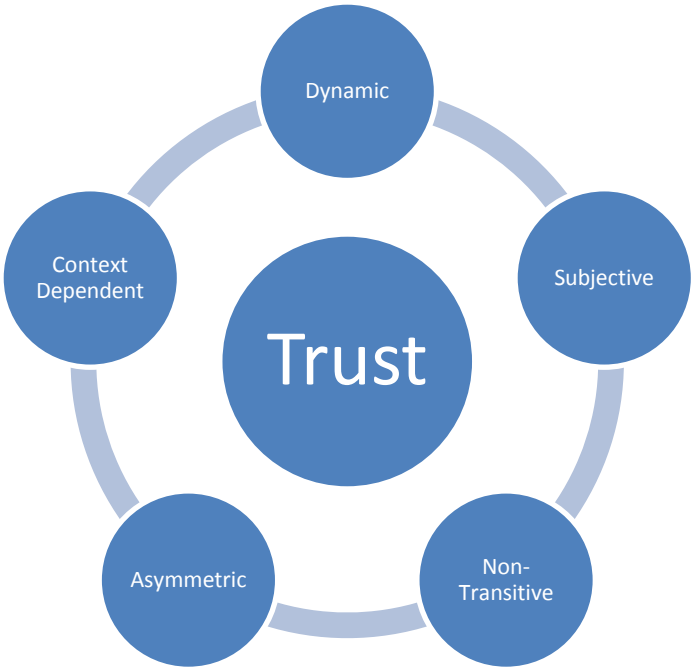

Figure 2.9: Properties of Trust metric [143]

In this work, a trust level is added to each node to make a fully distributive system. Trust based protocol calculates the trust values based on first hand observations as well as second hand observations [145-148]. Further, a secure end-to-end route free of the compromised node is searched.

Trust Evaluation (Trust Establishment and Trust Management) Procedure: The Trust evaluation is divided into following phases as Mohit Virendra et al. "Quantifying Trust in Mobile Adhoc Network" [149].

a) **Initiation and Monitoring Phase**: This is the phase when a new network is deployed. Here, no node has trust information about neighbor node or any other node. Therefore new node enters in the promiscuous mode and listens to the traffic. A new node initially is assigned a default trust value. It assumes that during this phase the possibility of the malicious node is minimum. Further, duration of this phase is kept minimum.

b) **Query and Evaluation phase**: In this phase, nodes evaluate their self trust. All the nodes exchange their trust value submitted to the neighbor node. Let us consider two nodes n_s and n_j in the wireless network then the trust of n_s on n_j is defined as follow. Here, $= \alpha_{1,} \alpha_2$ are weighing factors such that $\alpha_1 + \alpha_2 = 1$, $n_s T_{se}^{nd}$ is the ns self evaluated trust on nd and $n_s T_o^{nd}$ weighted sum of other nodes trust on nd evaluated by ns.

$$T\ n_{s,}\ n_j = \alpha_1 n_s T_{se}^{nd} + \alpha_2 n_s T_o^{nd} \quad (2.1)$$

c) **Updating Trust**: As long as the node remains in the transmission range of other node, its trust value is updated at regular time interval. Every node request trust value from its neighbor . In its reply, receiver node sends the trust value within minimum time interval. The sender node decrypts the trust value and verifies and validates. After all verifications, the sender node sends the acknowledgment packet to the neighbor confirming the receipt of trust value.

d) **Restructuring Phase**: A node enters in restructuring phase in the following condition.

- Trusted neighbor moves out of the radio range
- Trusted neighbor comes back in the radio range

e) **Reestablishment Phase:** If the node s monitoring node d finds that the node d as benign node as having the good trust value, but after some time node s finds the node d as malicious node because its trust value decreases below threshold value. Then node s and d enters in the reestablishment node and reestablishes their trust value.

Adnan Ahmed et al. in "A survey on trust based detection and isolation of malicious nodes in ad-hoc and sensor networks" [150] describes the work published on trust establishment schemes as shown in below figure 2.10.

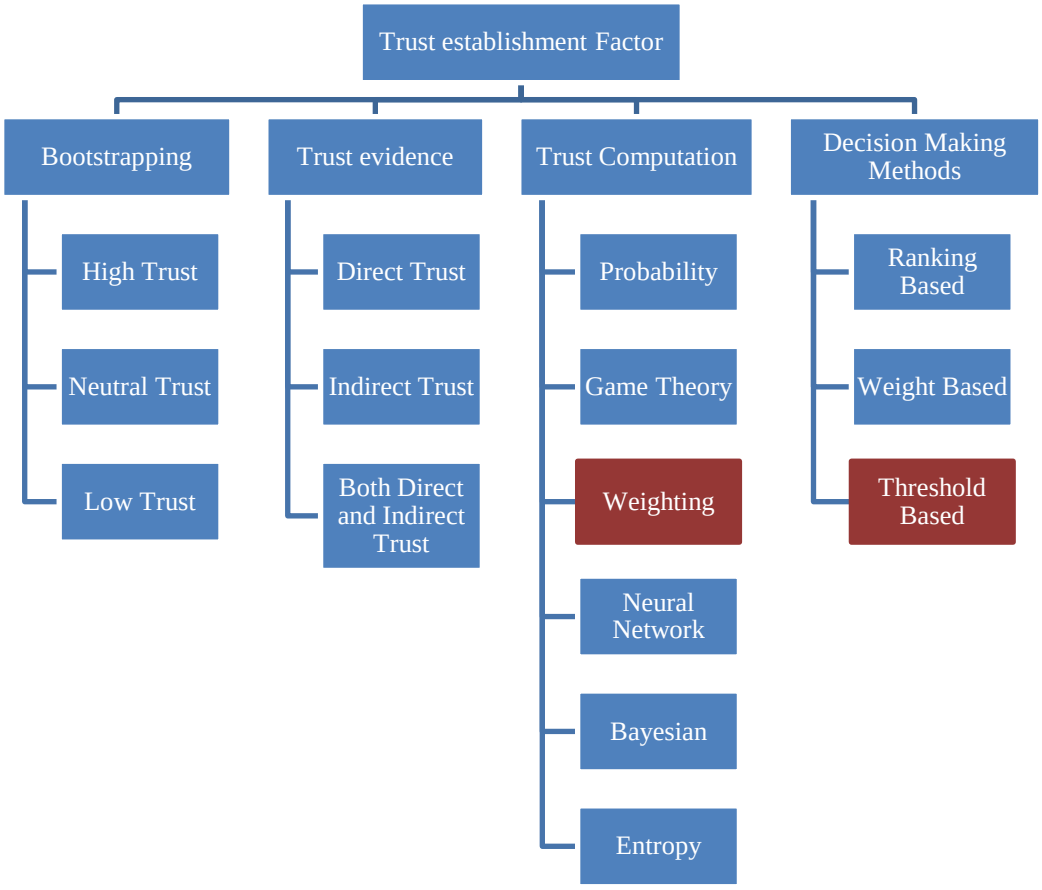

Figure 2.10: Various methods of Trust Computation [150]

This work uses trust evidence based on direct trust. Trust is computed based on weighting schemes and decisions are also take weight based. Some Existing Trust Based Models in MANETs that resembles the current work are discussed below [151-152]. Further, it also compares the performance, pro and cons of various trust based models.

Table 2.5: Existing weighted trust schemes

S.No.	**Existing work**	**Characteristics**
2006	Establishing Trust in Pure adhoc network [154]	Direct Observation
2006	Information Theoretic Framework of Trust Modeling and Evaluation for Ad Hoc Networks [155]	Direct observation on packet dropping rate

2006	Trust aware routing protocol (TARP) [156]	Direct observation on forwarding packet
2007	Trust and Recommendations in Mobile Ad Hoc Networks [157]	Direct Observation
2008	TSR: Trust-based Secure MANET Routing using HMMs [158]	Direct Observation
2013	Trust-Based Routing Mechanism in MANET: Design and Implementation (frAODV) [159]	Direct Observation
2014	Trust Based DSR Routing Protocol for Mitigating Cooperative Black Hole Attacks in Ad Hoc Networks [160].	Direct and indirect observations
2015	Agent-based trusted on-demand routing protocol for mobile ad-hoc networks [161]	Direct and indirect observations
2016	Enhancing security in MANETs through trust-aware routing [162]	Trust is based on neighbours feedback and past history of communication.
2017	Isolating Misbehaving Nodes in MANETs with an Adaptive Trust Threshold Strategy [163]	trust threshold in the routing protocol is decided based on connectivity, and average neighborhood trustworthiness
2018	A Comprehensive Trust-Aware Routing Protocol With Multi-Attributes for WSN [164]	Incorporates multiattributes of trust interms of communication, feedback etc.

Some examples of AODV based trust based models.

a) Collaborative Trust-Based Secure Routing against Colluding Malicious Nodes (T. Ghosh et al. [165]): It requires the existence of public key infrastructure. Here intermediate nodes are not allowed to send the reply control message so there is a delay in route discovery phase.

b) Trust-Embedded AODV (T-AODV) (T. Ghosh et al. [166]): It is an extension to Collaborative Trust-Based Secure Routing against Colluding Malicious Nodes and works only when there is a false accusation. It also assumes that all the nodes have same radio range.

c) Trust-Based Routing without Trust Infrastructure (Pirzada et al. [167]) : It does not requires the trust infrastructure. When the node transmits a packet, it overhears for the trust calculation.

d) Trust Establishment in Pure Ad-hoc Networks [168]: It is applicable to DSR, AODV and TORA. Here the node selects the most trustworthy next hop for communication. It may combine direct as well as indirect trust to fight with malicious behavior of node.

e) Opinion Based Trusted Routing Protocol – TAODV [169]: whenever node finds trustworthy node it authenticates it by signing its certificate.

f) Trust-Based Routing Mechanism in MANET: Design and Implementation (frAODV) [159]: Here, friendship based AODV routing algorithm is designed to add security. The friend list is created initially and distributed to the nodes. Each node maintains a list of friends and friendship value. The more friendship value indicates the more trustworthiness. During route discovery phase, the trust information is updated with the friend node only.

Generally, a trust factor based on the packet forwarding behaviour of neighbor can be used for detecting misbehaviour as previously presented in several literatures. For example, a trust factor of a node can be derived based on the number of forwarded packets at that neighboring node. But, by the same trust based detection logic, some of the neighbors those who were silent and not actively participated in communications will get low trust factor and will be wrongly identified as malicious. Because of this wrong identification, the link between source to destination will get broken at different locations

on their path because of this false identification of malicious node. In proposed work, it will overcome that problem and reduce the possibility of such false marking of non-malicious nodes as malicious nodes by introducing a Periodic Trust Handshake mechanism.

2.6 Summary

This chapter discusses the three types of attacks in detail: Jellyfish attack, Selfish node attack and Black hole attack. The main aim of Jellyfish attack is to drop the throughput of the network to nearly zero without much disobeying the protocol. Three major categories of jellyfish attack are: JF reorder attack, periodic dropping attack and delay variance attack. The next attack discussed here is Selfish node attack which is a type of passive attack where the selfish node does not participate in forwarding the data packets in order to save energy. And the last attack discussed is black hole attack in which malicious node first claims to have the shortest path. Once path having the malicious node is chosen for transmission, it intently drops all the data packets. At the end, the existing security solutions against these attacks are discussed in three categories: cryptographic security extensions, reputation based routing protocol and by adding trust metric to routing protocol. Cryptographic based routing solution is divided into asymmetric cryptographic solutions and symmetric cryptographic solutions. Asymmetric cryptographic solutions use trusted third party server to maintain the digital certificates of public keys. For example ARAN. As a third party server causes more overhead therefore symmetric cryptographic solution uses a single symmetric key to communication. For example SRP, SAODV, Ariadne. After studying and analyzing the existing cryptographic security enhancements on AODV routing protocol, it is observed that these protocols add a major overhead in terms of public and private keys and maintenance of various certificates. However they weakly protect them from certain active and passive attacks. In reputation based routing protocol, the cooperation among the nodes is enhanced by giving special incentive to the benign node. The incentive may be in form of remuneration or reputation. However, it also gives rise to difficulties such as identification of benign and malicious node, allocation of incentive after continuous observation of communication etc. Examples include CORE, Confidant etc. Reputations systems are more useful for detecting and avoiding the selfish node attack. Moreover, in

presence of selfish node attack, reputation system is vulnerable to false praise or false wrong doing. The false feedback may be given accidentally or intentionally. Moreover, there is also a need of system that also punishes the wrong feedback given by the node so as to avoid the presence of such nodes from the network. Whether it is cryptographic systems or reputation system, these schemes are either expensive or battery consuming. So they do not suite the model of MANETs. In third category, a special trust metric is embedded into the protocol. This work mainly focuses the addition of trust metric to existing AODV routing protocol so as to avoid attacks. This chapter discusses in detail the characteristics of trust metric and calculation of trust metric based on direct and indirect observations. It also presents the timeline of various common trust based extensions available in literature till now. Further it also presents some trust based extensions of AODV that have already been developed. But as per the study of various literatures it has been observed that some of the neighbors those who were silent and not actively participated in communications will get low trust factor and will be wrongly identified as malicious. Because of this wrong identification, the link between source to destination will get broken at different locations on their path because of this false identification of malicious node. In proposed work, it will overcome that problem and reduce the possibility of such false marking of non-malicious nodes as malicious nodes by introducing a Periodic Trust Handshake mechanism.

Chapter-3

Simulation Setup, Comparison of Attacks

The phases of network design are requirement analysis, flow analysis, logical design and physical design. Real network designing and analysis is a difficult process. Designing of a network is a difficult process due to external factors. External factors include government policies and regulations, technological services and products, organizational strategies etc. However, it can be made easy by building proper network model and choosing the right tool to evaluate the network. Tools to evaluate the network can be categorized into following categories: a) analytical tool b) simulation tool c) topological discovery tool d) topology generation tool [169]. Various simulation tools, characteristics and their advantages and disadvantages are described in table 3.1.

Analytical tool is used to design the network and to calculate various factors such as reliability. Simulation tool is used to study dynamic behavior of network. It models the actual or theoretical physical system and one can also manipulate the model in order to understand the characteristics of network. Table 3.1 describes various simulation tools.

The contents of this chapter have been peer reviewed and accepted for publication

[1] Bhawna Singla, A. K. Verma and L. R. Raheja, "An Evaluation on Selfish node Attack and Jellyfish Attacks Under AODV Routing Protocol" International Journal in Foundations of Computer Science & Technology March 2017, Volume 7, Number 2, p.p 15-28

[2] Bhawna Singla, A. K. Verma and L. R. Raheja, "A Comparative Analysis of Jellyfish Attacks and Black hole Attack with Selfish node Attack under AODV Routing Protocol" British Journal of Mathematics and Computer Science 2017, Volume 21, p.p.1-18

[3] Bhawna Singla, A. K. Verma and L. R. Raheja, " Performance Analysis of AODV in Presence of Attacks " WSEAS Transactions on Communication-2017, Volume 6, p.p.85-93 (Scopus indexed IF = .7).

[4] Bhawna Singla, A. K. Verma and L. R. Raheja, "Simulation of AODV under different attacks" Proceedings of International conference of Emerging Technologies-2014 pp.79-86 NCCE, Panipat.

Table 3.1: Comparison of various simulation tools [169]

S.No.	Tool Name	Characteristics	Advantages	Disadvantages
1	NS-2	Event driven object oriented simulator written in C++ and OTCL/TCL.	Many protocols already implemented New additions to protocol are possible Dynamic behavior can be visualized using NAM editor	Not advisable for large scale internet. Not suitable for real time applications.
2	Network Workbench	Discrete event simulator implemented in C++	It supports CSMA/CD collision backoff, Optimal route computation, reliable transport	The inbuilt implementation of protocol is very less.
3	Netsim	Mainly used for LAN	Source code is easily available and modifiable. GUI interface	Incorrect sequence of keys caused the simulation to crash.
4	MaryLand Simulator (MaRS)	It suits mainly to WAN and link state and distance vector routing	It provides flexible platform for implementation of network routing.	Its support for transport layer and the application layer is very limited.
5	Parallel/ Distributed	It uses parallel	Suitable for	

	ns (pdns)	simulated submodels	very large scale Network. IP addresses are also supported.	
6	Optimised Network engineering Tool (Opnet)	It is first commercial developed simulator	Extendible Large Customer base	High price Complex
7	Comet-III	Suitable for LAN, MAN, MAN Supports GUI	More realistic and accurate results.	Source code is not available New modules are difficult to add
8	REalistic And Large (Real)	Used for packet switched data network.	Extensible	Timers can not be reset using this method
9	GloMoSim	Specialized for wireless network	Wireless networks analysis	

Topology discovery tool helps to extract the topology information from the network and map then into geographical information. Topology generation tool helps to generate small and large topology in the network. Figure 3.1 further summarizes the common existing tools that are available nowadays in all the four categories: analytical tools, simulation tools, Topology discovery tool and Topology generation tool. Some tools are commercial while some tools are free of cost so that they can be used for educational purposes.

The results of the network in this work are taken with the help of NS2.35 simulator under Ubuntu platform. Survey of adhoc networks in MANETs [170] discusses various mobility models to determine how speed and direction changes within the reasonable time slot. This work uses random way point mobility model according to which a considerable pause time occurs between changes in destination and speed.

3.1 NS2.35 Simulator

NS is an event driven object oriented simulator written in C++ and OTCL/TCL as a front-end as shown in figure 3.2 [171-174]. It uses two languages because the simulation of protocols requires system programming language that can efficiently manipulate bytes, packets, routing protocols etc which is done in C++. Whereas research involves varying parameters and configurations which is done in OTCL/TCL. The OTCL script is used to initiate the event scheduler, set up the network topology, and tell traffic source when to start and stop sending packets through event scheduler. The scenes can be changed easily by programming in the OTCL script. When a user wants to make a new network object, he can either write the new object or assemble a compound object from the existing object library, and plumb the data path through the object.

When a new simulator object is created in TCL, initialization operation performs the following three functions

a) Initializes the packet format

b) Create the scheduler: Four schedulers present in NS are List scheduler, Heap scheduler, Calendar Queue scheduler, Real Time scheduler. The following is a list of the non-topology related simulator methods:

```
Simulator instproc now ;                          # return scheduler's notion of current time
Simulator instproc at args ;                      # schedule execution of code at specified time
Simulator instproc cancel args ;                  # cancel event
Simulator instproc run args ;                     # start scheduler
Simulator instproc halt ;                         # stop (pause) the scheduler
Simulator instproc flush-trace ;                  # flush all trace object write buffers
Simulator instproc create-trace type files src dst ;   # create trace object
Simulator instproc create_packetformat ;          # set up the simulator's packet format
```

c) Create a null agent: the null agent is created by the following call

```
set nullAgent_ [new Agent/Null]
```

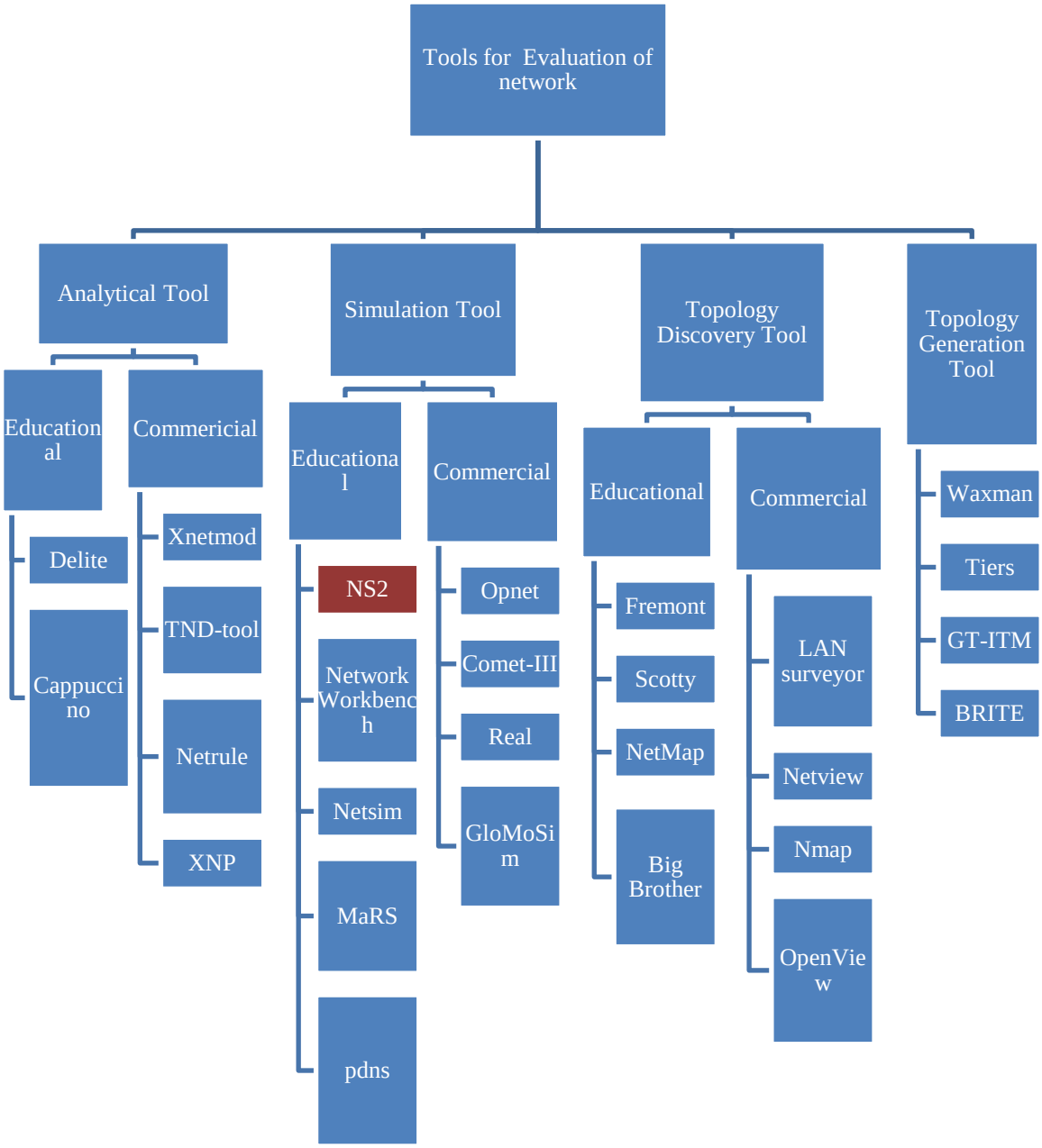

Figure 3.1: Various common tools for simulation of algorithms

3.1.1 Nodes , Links and Packet Forwarding

All the nodes contain the components like address, list of neighbors, list of agents, node type, routing module. The node in simulator is constructed as below,

```
set ns [new Simulator]
$ns node
```

Further, the class simplex-link makes the unidirectional link from one node to other.

$ns simplex-link (node 0) (node1) (bandwidth) (delay) (queue_type)

Different types of queues that are available in NS are Drop tail, Fair queuing, Stochastic fair queuing, Deficit round robin scheduling, Random early detection gateways, Weighted round robin scheduling etc. Further, a new header in packet can be added by the following steps

a) create a new structure defining new fields

b) create member function for the new fields

c) create a static class to perform OTCL/TCL linkage

d) edit ~ns/tcl/lib/ns-packet.tcl to enable new packet format

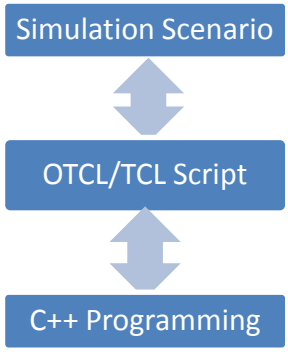

Figure 3.2: NS2 block diagram

3.1.2 Agents

It includes various fields which is assigned to packet before it is sent

addr_	node address of myself (source address in packets)
dst_	where packets is destined
size_	packet size in bytes (placed into the common packet header)
type_	type of packet (in the common header)
fid_	the IP flow identifier
prio_	the IP priority field flags_ packet flags
defttl_	default IP ttl value

The following OTCL code fragment creates a TCP agent and sets it up:

```
set tcp [new Agent/TCP] ;                    # create sender agent
$tcp set fid_ 2 ;                            # set IP-layer flow ID
$set sink [new Agent/TCPSink] ;              # create receiver agent
$ns attach-agent $n0 $tcp ;                  # put sender on node
```

```
$n0 $ns attach-agent $n3 $sink ;          # put receiver on node $n3
$ns connect $tcp $sink ;                  # establish TCP connection
set ftp [new Application/FTP] ;           # create an FTP source "application"
$ftp attach-agent $tcp ;                  # associate FTP with the TCP sender
$ns at 1.2 "$ftp start" ;                 #arrange for FTP to start at time 1.2 s
```

3.1.3. Mobile networking in NS

Following is a list of commands used in wireless simulations:

```
$ns_ node-config -addressingType   <usually flat or hierarchical>
-AdhocRouting                <Adhoc routing protocols such as DSR, AODV,
                              TORA    etc.>
-llType –                    <link layer>
-macType                     <mac type sch as 802.11>
-propType                    <propagation model such as two way ground>
-ifqType                     <interface queue such as drop tail as mentioned previously>
-ifqLen                      <interface queue length>
-phyType                     <network interface such as wireless>
-antType                     <antennae such as omni antennae>
-channelType                 <channel type such as wireless>
-topoInstance                <topological information instance>
-wiredRouting                <turning the wired routing on or off>
-mobileIP                    <tuning the mobile IP on or off>
-energyModel                 <energy model type>
 -initialEnergy              <specified in Joules>
-rxPower                     <specified in W>
-txPower                     <specified in W>
-agentTrace                  <tracing at agent level, may be on or off>
-routerTrace                 <tracing at router level, may be on or off>
-macTrace                    <tracing at mac level, may be on or off>
-movementTrace               <mobile node movement logging, may be on or off>
```

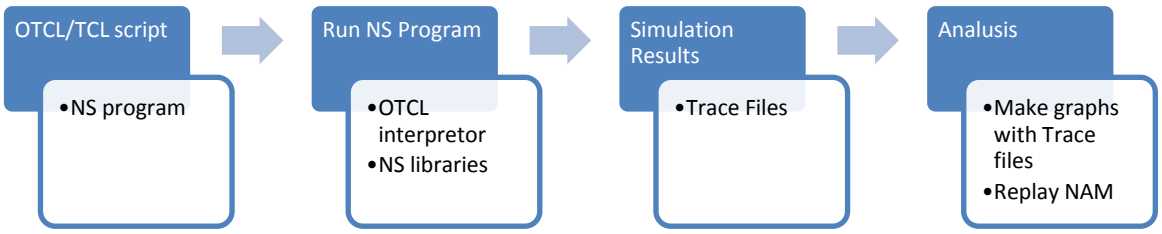

Figure 3.3: NS2 components

3.1.4 NAM

NAM produces the trace files having the following fields

Event	Time	From Node	To Node	Pkt Type	Pkt Size	Flags	Fid	Source Address	Destination address	Sequence no.	Pkt Id

Figure 3.4: Various fields of trace file

Here, the event type may be enqueue packet, dequeue packet, receive packet and drop packet.Time field denotes the time at which event occurs. From Node is the input node of the link at which event occurs. To node is the output node at which the event occurs. Packet type is the type of packet such as TCP, UDP etc. The data of trace files are analyzed by Xgraph in this work.

3.2 Simulation of Attacks Under NS2 Simulator

In this section, the three attacks i) Jellyfish attack ii) Selfish node attack iii) Black hole attack are implemented to better understand their impact on AODV routing protocol. First of all the pseudo code of attack is presented in the next section. Then the performance of AODV in presence of attacks is presented and compared.

3.2.1 Pseudo Code of Attacks

The theoretical behaviour of attacks, discussed in chapter 2, can be better understood by implementing these attacks on AODV (discussed in section 1.5, the routing protocol chosen for the study in this work) routing protocol. The following Pseudo Code (page number 61-63) in figure 3.5 is developed and added in the packet forwarding stage of existing AODV code for simulating malicious behaviour. After the three attacks are simulated on AODV routing protocol, it clears the picture that to how these attacks

disturb the performance of routing protocol and to how much extent these attacks are dangerous.

```
Forward(Pkt, Delay) {
if ( ttl=0 ) {
            drop(Pkt);
            return;
               }
 if (pkt is addressed to this node) {
                                        recv(pkt);
                                        return;
                                        }
if (pkt  is a AODV broadcast) {
                                  scheduleTransmission(pkt,delay)
                                  //The Attacks on AODV pkt is not implemented
                                  } else {
     // here it is a data packet which needs to be   forwarded
    If  (AttackMode= none)  {
                scheduleTransmission(pkt,delay)
                                   else if (AttackMode= JellyfishReorder) {
                                   //If dest  is me then process the packet normally
                                  if ( pkt addressed to this node) {
                        scheduleTransmission(pkt,delay)
                                  } else {
                                   //here we are imposing reorder
                      //by scheduling the packets at random time
                                 delay= JellyfishReorderLimit * Rand() ;
           scheduleTransmission(pkt,delay)
                                  }

    else if (AttackMode= JellyfishPeriodicDropping) {
           //If dst is me then process the packet normally
```

```
   if ( pkt addressed to this node) {
                    scheduleTransmission(pkt,delay)
            } else {
              // imposing periodic packet dropping
 //by scheduling the packets at random time
              if (JellyfishAttackProbability > Rand()) {
                  scheduleTransmission(pkt,delay)
               } else {
                  //Malicious Dropping
                  drop(pkt)}   }

else if (AttackMode= JellyfishDelayVariance) {
        //If dest is me then process the packet normally
           if ( pkt addressed to this node) {
          scheduleTransmission(pkt,delay)
           } else {
       //scheduling the packets with high delay
                   delay= JellyfishAttackDelay+ Rand() ;
    scheduleTransmission(pkt,delay)
           }

 else if (AttackMode= SelfishBehaviour) {
          //If dest is me then process the packet normally
           if ( pkt addressed to this node) {
          scheduleTransmission(pkt,delay)
           } else {
              //here we are behaving selfishly
          MaliciousDrop(pkt);
           }

 else if (AttackMode= Black hole) {
          //If dest is me then process the packet normally
```

```
if ( pkt addressed to this node) {
scheduleTransmission(pkt,delay)
} else {
   //here we are behaving selfishly
MaliciousDrop(pkt);
}    }}
```

Figure 3.5: Forward Packet Function

In addition to the above function, for black hole attack, other functions also implemented for generating fake replies to the RREQ messages as shown in figure 3.6.

```
OnRecieveRReq( Pkt ) {
if (AttackMode= Black hole) {
 // send fake reply with lesser hop
  sendFakeReply();
  //drop the original request
  drop(Pkt);
else
        if (CurrentNodeIsDestination) {
                //send normal reply
                sendRReply()
      }else {
        //forward pkt normally
             Forward(Pkt)}}
```

Figure 3.6: OnRecieveRReq Function

3.2.2 Methodology

The attacks are implemented on the AODV code of NS2 by the following changes.

a) **Changes Made in AODV.h**: The additional function definitions for simulating attacks and the variables that will be bound with TCL are declared in AODV.h. By using the variables from a TCL simulation code, we can control the behavior of the routing agent.

b) Changes Made in AODV.cc: The actual code of the additional function definitions for simulating attacks was implemented in AODV.cc. And here the new interfaces to the code through the control variables that will be bound with TCL are written here. By setting the variables from a TCL simulation code, one can control the behavior of the routing agent.

The following main functions were also modified for simulating attacks

a) The Function AODV::command(..): Here the interface to the newly added functionalities are provided. It means one can set some of the variables of C++ code from the TCL simulation script through the interfaces provided in this function.

b) The Function AODV::AODV(..): In the constructor section of the AODV code, the code needed for binding of new control variables is added.

c) The function AODV::recvRequest(..): In this function, the malicious fake route reply code for black hole attack is implemented. With respect to the value of a control variable "AttackType", the AODV routing agent will behave normal or malicious.

d) The function AODV::forward(..): In this function, the code for different attacks such as Jellyfish Reorder Attack, Jellyfish Periodic Dropping Attack Jellyfish Delay Variance Attack, selfish node attack and Black Hole Attack were implemented. With respect to the value of a control variable "AttackType", the AODV routing agent will behave normal or do a particular attack.

After the modifications on AODV.h and AODV.cc, the new version of NS2 is compiled to incorporate the modified version of AODV routing agent. Now the modified version of AODV routing agent can be used in a TCL simulation code. And the functionality of the AODV agent can be controlled by setting up the suitable value in control variable or a using appropriate AODV initialization function that is newly added in AODV::command(..) section.

3.2.3 Simulation Parameters

Common Parameters: In simulation, the following common parameters are shown in table 3.2 is used, while setting up the network.

Table 3.2: Simulation parameters

Parameters	Values
Topographical Area (m*m)	1800 X 500
Mobility	20m/s
Pause Time	20s
Total SimulationTime	100s
Routing Protocol	AODV
MobilityModel	RandomWaypoint
Channel Model	WirelessChannel
Propagation Model	TwoRayGround
PhyModel	WirelessPhy
MacModel	802_11
AntennaModel	OmniAntenna
Queue	DropTail-PriQueue
Queue Length	50

Traffic Parameters: The following parameters shown in table 3.3 are used in setting up the TCP flows with some periodic data.

Table 3.3: Parameters for setting up TCP flow

Transport Agent	TCP
No Flows	10
Traffic Type	CBR
Packet Size	1KB
Interval	100ms
Rate	10KB

Variable Parameters: The following parameters shown in table 3.4 are used as variables for analyzing the impact of the different attacks on different condition.

Table 3.4: Parameters for analyzing the impacts of attacks

Parameters	Values
Attacking Nodes	5, 10, 15 and 20
Total Nodes	40, 50,60
Simulated Attacks	Jellyfish Reorder Attack Jellyfish Periodic Dropping Attack Jellyfish Delay Variance Attack Selfish Node Attack Black hole Attack

3.3 Results showing the Impact of Attacks on AODV routing protocol

3.3.1. Analysis –I - Network Size Vs Performance

The performance of AODV protocol is analyzed by varying the size of the network as 40, 50 and 60, represented by nodes in the table, in the NS2.35. The following trace files are observed. Here PDF stands for packet delivery fraction, NRL is the normalized routing load, EED is the end-to-end delay, overhead is the packet overhead, SDDropped is the number of dropped packet at the network layer, Throughput is the ratio of received packet and sent packet, MACLoad is the overhead at the MAC layer, ConsEnergy is the consumed energy of battery, MALDropped is the maliciously dropped packet at the network layer, MACDropped is the dropped packet at the MAC layer , sent is the number of sent packet and Receive is the number of received packet.

Table 3.5: Trace file of AODV without attack

Nodes	PDF	NRL	EED	Overhd	SDDropped	Throughput	MAC Load	ConsEnergy	MalDropped	MAC Dropped	Sent	Received
40	81.00	8.79	0.39	12470.	333.00	177.45	27.91	10.04	0.00	7609.0	1751.0	1418.0
50	83.00	8.43	0.36	11903.	290.00	170.07	24.78	9.13	0.00	11416.	1702.0	1412.0
60	74.40	14.37	0.72	15862.	380.00	121.47	39.87	10.22	0.00	28864.	1484.0	1104.0

Table 3.6: Trace file of AODV with Jellyfish reorder attack

Nodes	PDF	NRL	EED	Overhd	SDDropped	Throughput	MAC Load	ConsEnergy	MalDropped	MAC Dropped	Sent	Received
40	75.30	8.12	0.99	9181.0	371.00	127.89	24.43	6.85	0.00	5481.0	1502.0	1131.0
50	78.30	7.46	0.89	8916.0	331.00	133.51	21.46	6.77	0.00	8059.0	1526.0	1195.0
60	71.20	11.98	0.98	12448.	421.00	113.50	33.27	8.69	0.00	20026.	1460.0	1039.0

Table 3.7: Trace file of AODV with Jellyfish periodic dropping attack

Nodes	PDF	NRL	EED	Overhd	SDDropped	Throughput	MAC Load	ConsEnergy	MalDropped	MAC Dropped	Sent	Received
40	54.30	8.87	0.17	6890.0	654.00	97.22	25.39	4.65	461.00	4354.0	1431.0	777.00
50	66.50	9.39	0.31	8709.0	466.00	102.68	24.94	5.48	386.00	9300.0	1393.0	927.00
60	47.80	18.79	0.27	11347.	660.00	65.19	46.60	6.17	601.00	22380.	1264.0	604.00

Table 3.8: Trace file of AODV with Jellyfish delay variance attack

Nodes	PDF	NRL	EED	Overhd	SDDropped	Throughput	MAC Load	ConsEnergy	MalDropped	MAC Dropped	Sent	Received
40	70.50	8.15	1.73	7835.0	402.00	99.83	24.34	5.81	0.00	4391.0	1363.0	961.00
50	81.40	8.03	1.29	10120.	289.00	142.20	23.29	7.56	0.00	8791.0	1550.0	1261.0
60	60.40	12.98	2.06	9630.0	486.00	66.81	34.38	6.06	0.00	16852.	1228.0	742.00

Table 3.9: Trace file of AODV with Selfish node attack

Nodes	PDF	NRL	EED	Overhd	SDDropped	Throughput	MAC Load	ConsEnergy	MalDropped	MAC Dropped	Sent	Received
40	45.80	7.55	0.05	4576.0	718.00	75.99	20.72	2.92	551.00	2741.0	1324.0	606.00

50	47.50	10.64	0.10	7172.0	746.00	91.22	27.20	4.54	695.00	8533.0	1420.0	674.00
60	38.00	17.35	0.14	8501.0	800.00	62.95	42.71	4.89	600.00	15961.	1290.0	490.00

Table 3.10: Trace file of AODV with Black hole attack

Nodes	**PDF**	**NRL**	**EED**	**Overhd**	**SDDropped**	**Throughput**	**MAC Load**	**ConsEnergy**	**MalDropped**	**MAC Dropped**	**Sent**	**Received**
40	0.40	655.75	0.08	2623.0	1016.0	0.32	1906.7	1.48	427.00	1564.0	1020.0	4.00
50	1.80	163.94	0.11	2951.0	1002.0	0.81	438.83	1.66	462.00	2377.0	1020.0	18.00
60	7.10	63.51	0.17	4763.0	986.00	10.93	171.48	2.64	804.00	7035.0	1061.0	75.00

The following graph in figure 3.7 shows the impact of different attacks in terms of total data packets sent at application source. As shown in the line graph, under the presence of black hole Attack the application source itself can not able to send much. selfish node attack seems to be causing a little bit higher impact than all the Jellyfish Attacks. With respect to the increase of number of attackers, the performance decreases.

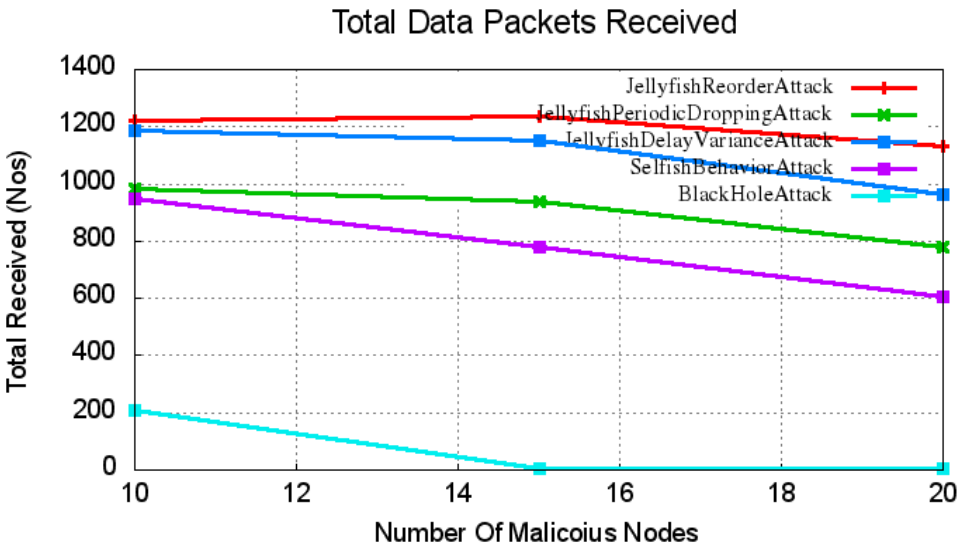

Figure 3.7: Comparison of Attackers Vs Sent Data Packets in presence of attacks on AODV.

The graph in figure 3.8 shows the impact of different attacks in terms of total data packets received at application destination. As shown in the line graph, under the presence of black hole Attack the application destination can not able to receive much. selfish node attack seems to be causing a little bit lower higher than all the Jellyfish Attacks. With respect to the increase of no of attackers, the performance decreases.

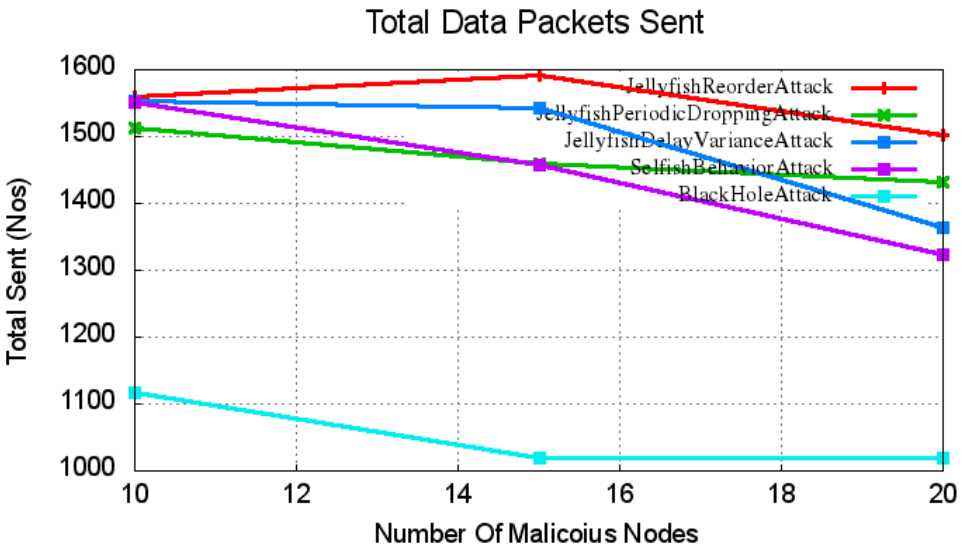

Figure 3.8: Comparison of Attackers Vs Received Data Packets in presence of attacks on AODV.

The graph in figure 3.9 shows the impact of different attacks in terms of data packets dropped at source and destination. It signifies the packets dropped at the the application layer. As shown in the line graph, black hole Attack caused much packet dropping at the application layer. selfish node attack seems to be causing a little bit higher impact than all the Jellyfish Attacks. With respect to the increase of no of attackers, the performance slightly decreases.

The graph in figure 3.10 shows the impact of different attacks in terms of maliciously dropped data packets at routing layer. As shown in the line graph, except Jellyfish Reorder Attack and Jellyfish Delay Variant attack, all the other attacks maliciously dropping packets at routing layer. Of course, conceptually, Jellyfish Reorder Attack and Jellyfish Delay Variant attack will not drop any packet at routing layer; but only affect the packet transmission/forwarding in different way. The selfish node causes a little bit of high data packet drop at routing layer. With respect to the increase of no of attackers, the malicious drop at routing layer is getting increase considerably.

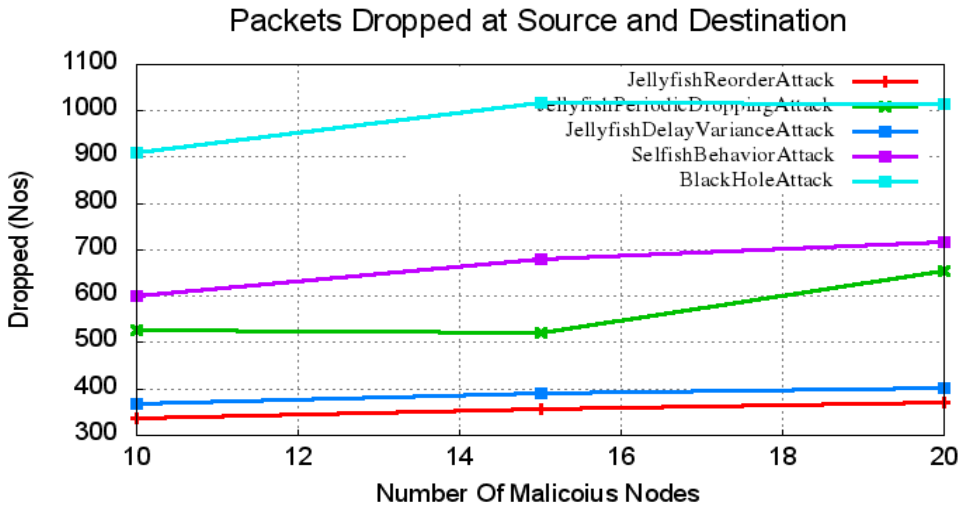

Figure 3.9: Comparison of Attackers Vs Dropped packet at the application layer in presence of attacks on AODV.

The graph in figure 3.11 shows the impact of different attacks in terms of average achieved throughput of TCP flows. As shown in the line graph, black hole Attack caused much packet loss so that the throughput was very lower than all other attacks. Next to black hole attack, selfish node attack seems to be causing a little bit higher impact than all the Jellyfish Attacks. With respect to the increase of no of attackers, the throughput is getting decreased considerably. As the throughput is the ratio od number of received packets successfully at the destination and black hole attack causes the drop of number of packets therefore throughput considerably drops in case of black hole attack and selfish

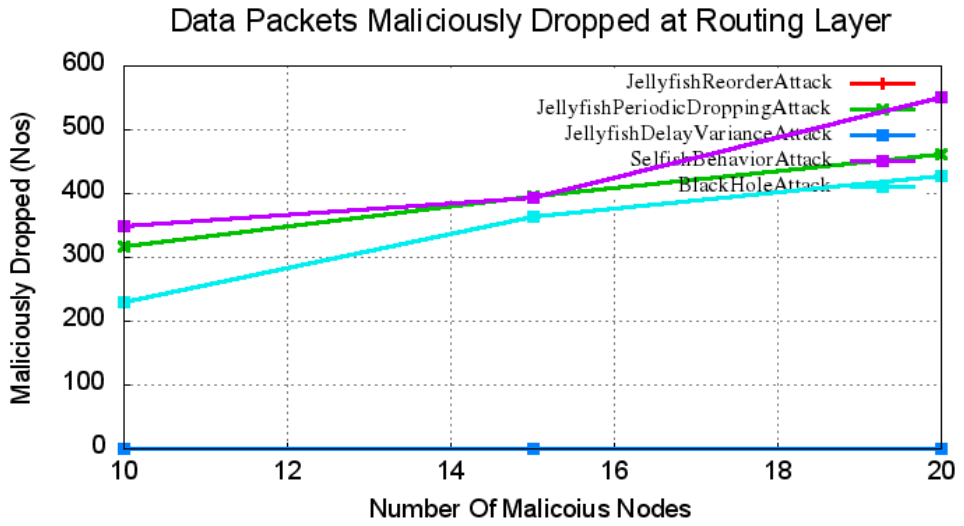

Figure 3.10: Comparison of Attackers Vs Maliciously Dropped at Routing Layer in presence of attacks on AODV.

node attack also causes the non-forwarding of packets which is the reason for drop of throughput.

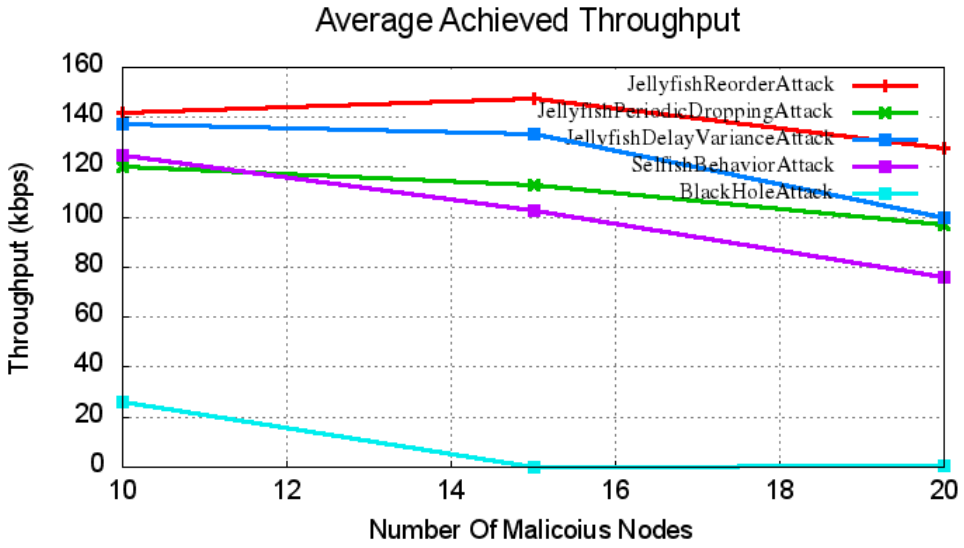

Figure 3.11: Comparison of Attackers Vs Throughput in presence of attacks on AODV.

The graph in figure 3.12 shows the impact of different attacks in terms of Packet Delivery Fraction (PDF) of TCP flows. As shown in the line graph, black hole attack caused much packet loss so that the PDF was very lower than all other attacks. Next to black hole attack, selfish node attack seems to be causing a little bit higher impact than all the Jellyfish Attacks. With respect to the increase of no of attackers, the performance decreases.

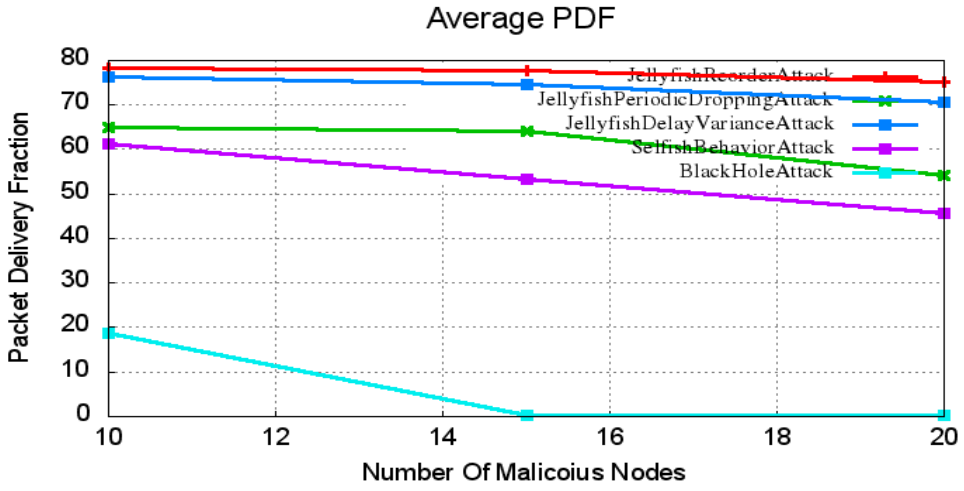

Figure 3.12: Comparison of Attackers Vs Dropped PDF in presence of attacks on AODV.

The graph in figure 3.13 shows the impact of different attacks in terms of End-to-end Delay (EED) of TCP flows. As shown in the line graph, black hole attack and selfish node attack seems to be providing lower EED than all other Jellyfish Attacks – but certainly, it does not mean that black hole attack and selfish node attack are improving the performance or the network. With respect to the increase of number of attackers, the performance is affected with respect to the nature of attack.

The low EED under these two attacks are due to a strange fact that these two attacks make disconnection in TCP flows and since the packets are not at all forwarded to any further nodes, indirectly it reduces the message overhead network and reduced bandwidth usage otherwise it will be consumed by the forwarded data packets. So, the flows that were unaffected by black hole attack and selfish node attack (the connections where there are no neighboring attack nodes) utilize that extra bandwidth and gains some performance in term of some metrics.

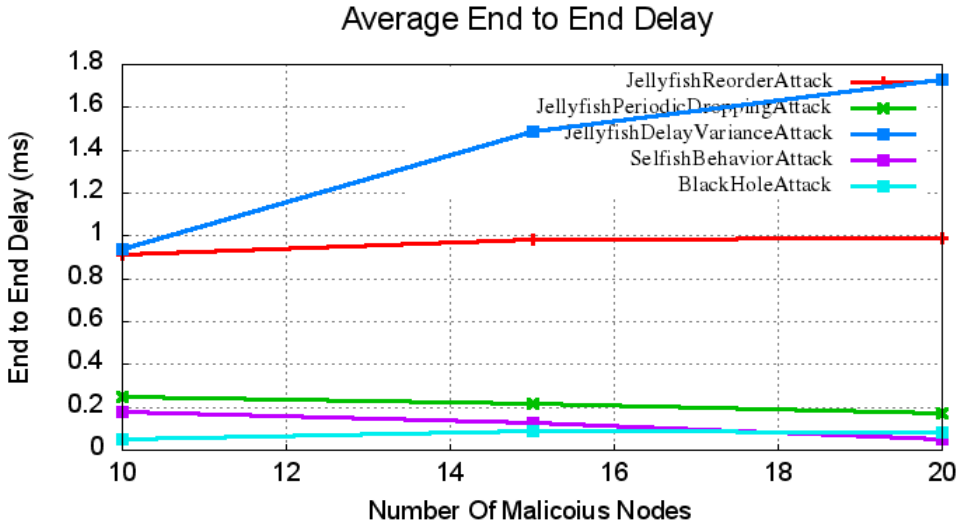

Figure 3.13: Comparison of Attackers Vs End-to-end Delay in presence of attacks on AODV.

The graph in figure 3.14 shows the impact of different attacks in terms of consumed battery energy. As shown in the line graph, in the presence of black hole attack and selfish node attack the battery consumption is lesser than all other Jellyfish attacks – but certainly it does not mean that black hole attack and selfish node attack are improving the performance in terms of energy consumption. The low energy consumption under this two attacks are due to a strange fact that these two attacks makes disconnection in TCP flows and since the packets are not at all forwarded to any further nodes, indirectly it is reduce the battery consumption at the other nodes otherwise it will be consumed for

forwarding the data packets. So, the nodes that were unaffected by black hole attack and selfish node attack (where there are no neighboring attack nodes) preserves some battery power. Understanding this strange fact requires a better visualization of the whole network scenario.

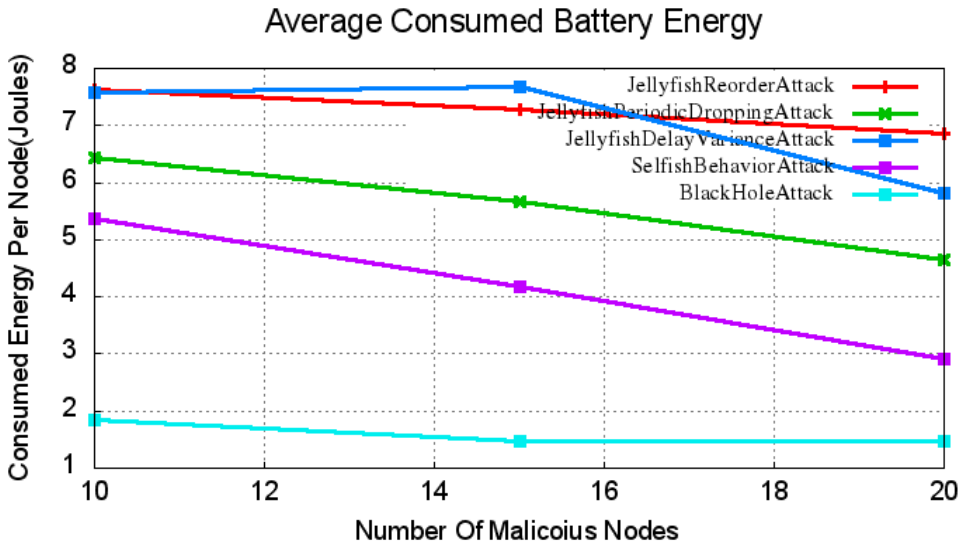

Figure 3.14: Comparison of Attackers Vs Battery Energy in presence of attacks on AODV.

3.2.3 Analysis –II – Malicious Nodes Vs Performance

In the following analysis the total number of nodes in the network is kept as 40 and among them, the number of malicious nodes was varied as 10, 15 and 20 represented by MaNodes in the trace files. And the impact is measured using different metrics.

Table 3.11: Trace file of AODV with Jellyfishreorderattack and variable malicious node

MaNodes	PDF	NRL	EED	Overhd	SDDropped	Throughput	MAC Load	ConsEnergy	MalDropped	MAC Dropped	Sent	Received
10	78.30	7.82	0.91	9543.0	338.00	141.62	24.42	7.64	0.00	5751.0	1559.0	1221.0
15	77.60	7.17	0.98	8853.0	357.00	147.61	22.49	7.28	0.00	4579.0	1592.0	1235.0
20	75.30	8.12	0.99	9181.0	371.00	127.89	24.43	6.85	0.00	5481.0	1502.0	1131.0

Table 3.12: Trace file of AODV with Jellyfish periodic dropping attack and variable malicious node

MaNodes	PDF	NRL	EED	Overhd	SDDropped	Throughput	MAC Load	ConsEnergy	MalDropped	MAC Dropped	Sent	Received
10	65.10	8.46	0.25	8337.0	527.00	120.45	25.29	6.44	317.00	4971.0	1512.0	985.00
15	64.20	8.71	0.22	8169.0	522.00	112.78	25.09	5.66	395.00	5170.0	1460.0	938.00
20	54.30	8.87	0.17	6890.0	654.00	97.22	25.39	4.65	461.00	4354.0	1431.0	777.00

Table 3.13: Trace file of AODV with Jellyfish Delay variance attack and variable malicious node

MaNodes	PDF	NRL	EED	Overhd	SDDropped	Throughput	MAC Load	ConsEnergy	MalDropped	MAC Dropped	Sent	Received
10	76.30	8.32	0.94	9862.0	368.00	137.39	25.45	7.59	0.00	6049.0	1553.0	1185.0
15	74.60	8.89	1.49	10236.	391.00	133.26	26.30	7.68	0.00	6218.0	1542.0	1151.0
20	70.50	8.15	1.73	7835.0	402.00	99.83	24.34	5.81	0.00	4391.0	1363.0	961.00

Table 3.14: Trace file of AODV with Selfish node attack and variable malicious node

MaNodes	PDF	NRL	EED	Overhd	SDDropped	Throughput	MAC Load	ConsEnergy	MalDropped	MAC Dropped	Sent	Received
10	61.20	7.26	0.18	6881.0	602.00	124.94	21.20	5.36	349.00	4075.0	1550.0	948.00
15	53.40	8.30	0.13	6456.0	680.00	102.56	21.93	4.17	394.00	4144.0	1458.0	778.00
20	45.80	7.55	0.05	4576.0	718.00	75.99	20.72	2.92	551.00	2741.0	1324.0	606.00

Table 3.15: Trace file of AODV with Black hole attack and variable malicious node

MaNodes	PDF	NRL	EED	Overhd	SDDropped	Throughput	MAC Load	Cons Energy	Mal Dropped	MAC Dropped	Sent	Received

10	18.60	14.16	0.05	2946.0	910.00	26.21	39.55	1.84	229.00	1698.0	1118.0	208.00
15	0.30	909.33	0.09	2728.0	1017.0	0.10	2425.0	1.46	364.00	1511.0	1020.0	3.00
20	0.40	655.75	0.08	2623.0	1016.0	0.32	1906.7	1.48	427.00	1564.0	1020.0	4.00

Here we see the analytic results of comparison of different attacks with AODV (it means performance without any attack and it is studied with respect to different network size. In the following analysis the total number of nodes in the network is varied as 40, 50 and 60 and among them, the number of malicious nodes kept as 20 and the impact is measured using different metrics.

The graph in figure 3.15 shows the impact of different attacks in terms of total data packets sent at application source. As shown in the line graph, under the presence of black hole attack the application source itself can not able to send much. selfish node attack seems to be causing almost equal impact like all the Jellyfish Attacks. But even without the presence of any attack, AODV performs good and able to send much data packets. With respect to the increase of no of nodes in the network, the performance decreases in most of the cases.

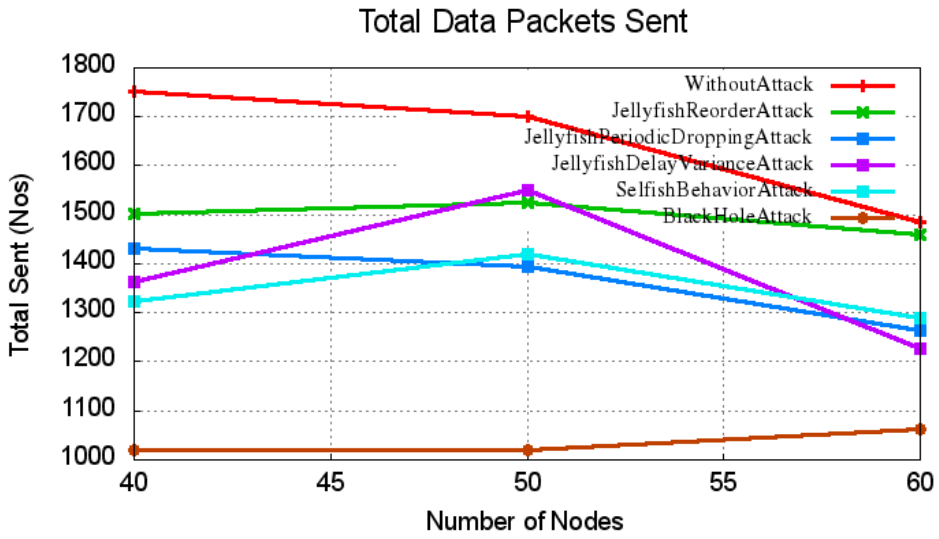

Figure 3.15: Comparison of Network Size Vs Sent Packets on AODV with varying malicious node.

The graph shows in figure 3.16 the impact of different attacks in terms of total data packets received at application destination. As shown in the line graph, under the presence of black hole attack the application destination can not able to receive much. Next to black hole attack, selfish node attack seems to be causing much impact

than all the Jellyfish Attacks. But even without the presence of any attack AODV performs good and able to send much data packets. With respect to the increase of number of nodes in the network, the performance decreases in most of the cases.

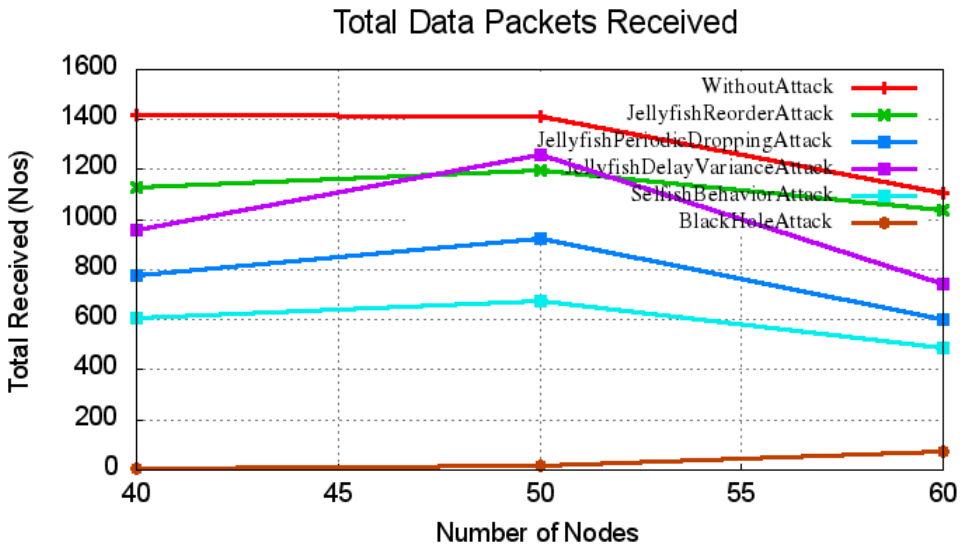

Figure 3.16: Comparison of Network Size Vs Received Packets on AODV with varying malicious node.

The graph in figure 3.17 shows the impact of different attacks in terms of data packets dropped at source and destination. It signifies the packets dropped at the application layer. As shown in the line graph, black hole attack, selfish node attack and Jellyfish Periodic Packet Dropping Attacks were causing much packet drop at the application layer. The other two types of Jellyfish Attacks also causing a little packet drop at the application layer. But without the presence of any attack AODV performs good and dropping less packets at the application layer packets at the application layer.

The graph in figure 3.18 shows the impact of different attacks in terms of maliciously dropped data packets at routing layer. As shown in the line graph, except Jellyfish Reorder Attack and Jellyfish Delay Variant attack, all the other attacks maliciously dropping packets at routing layer. Of course, conceptually, Jellyfish Reorder Attack and Jellyfish Delay Variant attack will not drop any packet at routing layer; but only affect the packet transmission/forwarding in different way. The selfish node causes a little bit of high data packet drop at routing layer. With respect to the increase of no of nodes in the network, the malicious dropping increasing a little bit.

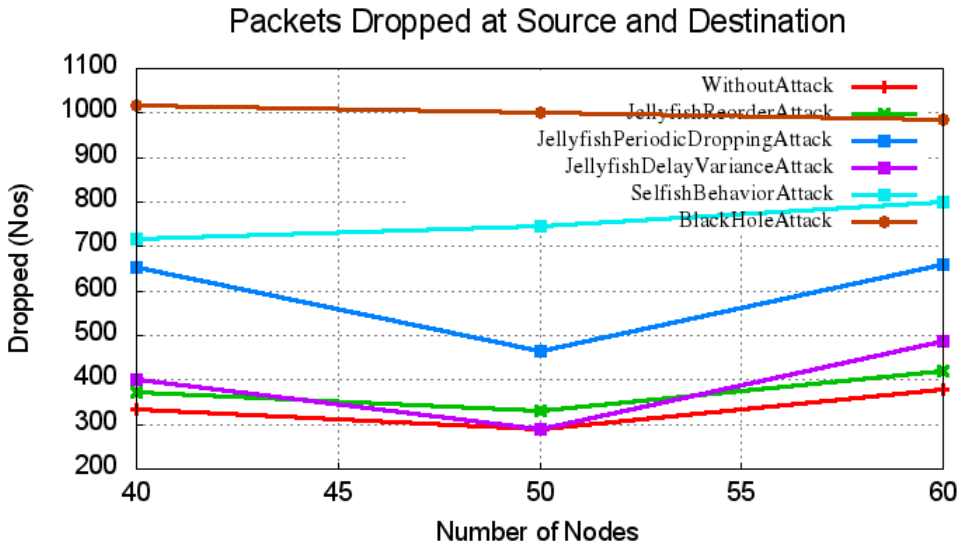

Figure 3.17: Comparison of Network Size Vs Packets Dropped At The application layer on AODV with varying malicious node.

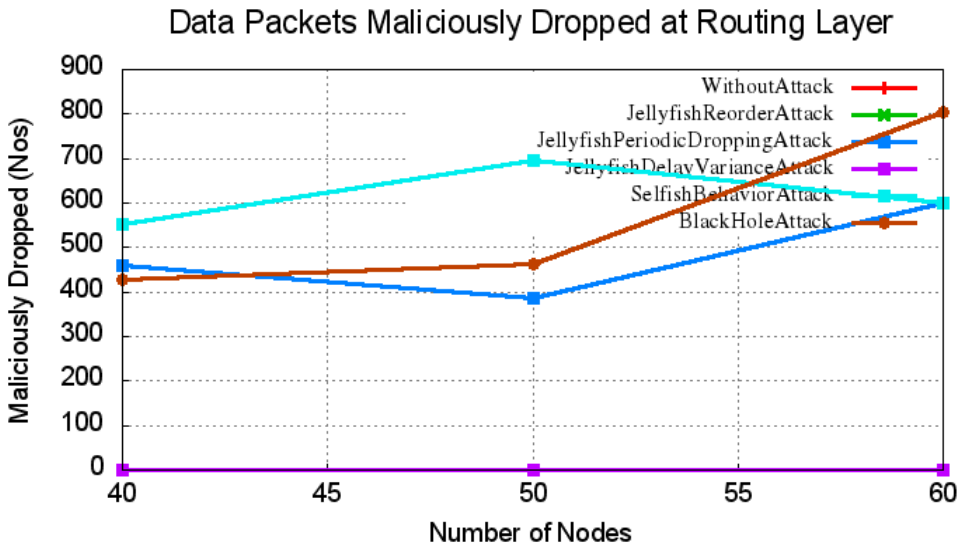

Figure 3.18: Comparison of Network Size Vs Malicious Drops at Routing Layer on AODV with varying malicious node.

The graph in figure 3.19 shows the impact of different attacks in terms of average achieved throughput of TCP flows. As shown in the line graph, black hole attack caused much packet loss so that the throughput was very lower than all other attacks. Next to

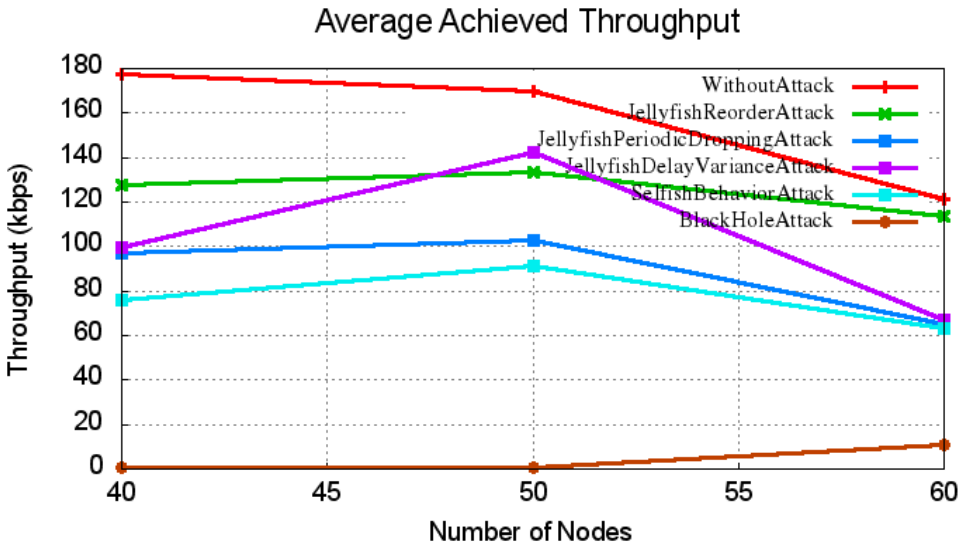

Figure 3.19: Comparison of Network Size Vs Throughput on AODV with varying malicious node.

black hole attack, selfish node attack seems to be causing a little bit higher impact than all the Jellyfish Attacks. But without the presence of any attack AODV performs good and provided the highest throughput. With respect to the increase of number of nodes in the network, the throughput decreases considerably.

The graph in figure 3.20 shows the impact of different attacks in terms of Packet Delivery Fraction (PDF) of TCP flows. As shown in the line graph, black hole attack caused much packet loss so that the PDF was very lower than all other attacks. Next to black hole attack, selfish node attack seems to be causing a little bit higher impact than all

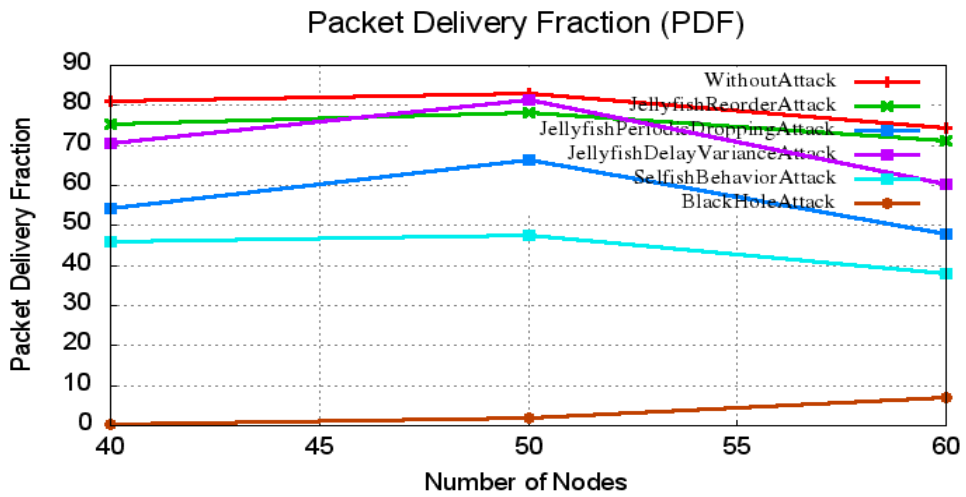

Figure 3.20: Comparison of Network Size Vs PDF on AODV with varying malicious node.

the Jellyfish Attacks. But without the presence of any attack AODV performs good and provided highest PDF. With respect to the increase of no of nodes in the network, the performance getting decreased in most of the cases. But in the case of Black hole attack, with respect to the increase of number of nodes in the network, the performance getting increased because with the high number of nodes, there were chances for developing alternate path that may avoid malicious nodes in it.

The graph in figure 3.21 shows the impact of different attacks in terms of End-to-end Delay (EED) of TCP flows. With respect to the increase of no of nodes in the network, the performance getting decreased. As shown in the line graph, black hole attack and selfish node attack seems to be providing lower EED. The low EED under this two attacks are due to a strange fact that these two attacks make disconnection in TCP flows and since the packets are not at all forwarded to any further nodes, indirectly it is reduce the message overhead in the network and reduced bandwidth usage otherwise it will be consumed by the forwarded data packets. So, the flows that were unaffected by black hole attack and selfish node attack (the connections where there is no neighboring attack nodes) utilizes that extra bandwidth and gains some performance in term of some metrics.

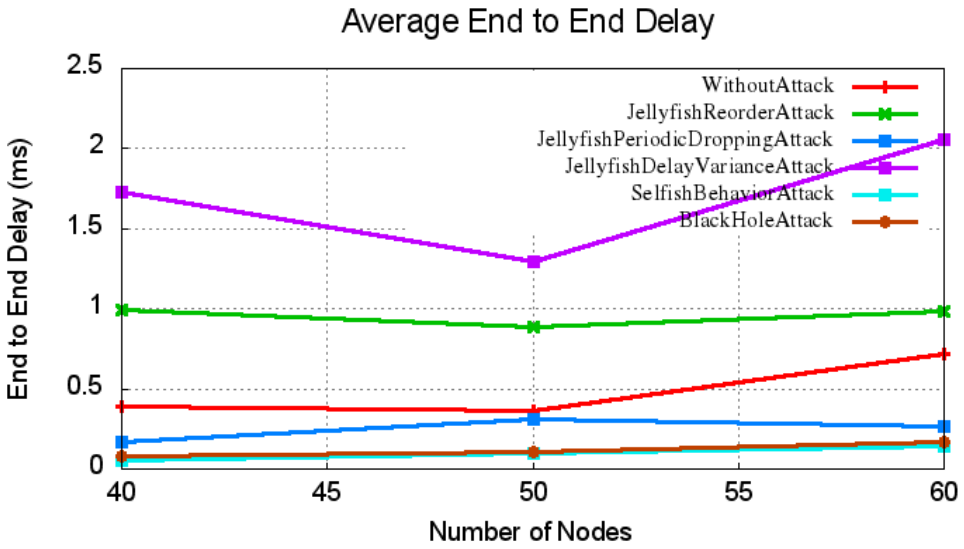

Figure 3.21: Comparison of Network Size Vs End-to-end Delay on AODV with varying malicious node.

The graph in figure 3.22 shows the impact of different attacks in terms of consumed battery energy. As shown in the line graph, in the presence of all the kinds of Attack the battery consumption is lesser than AODV (without attack). The low energy consumption under attacks are due to a strange fact that these attacks make disconnection

in TCP flows and since the packets are not at all forwarded to any further nodes, indirectly it is reduce the battery consumption at the other nodes otherwise it will be consumed for forwarding the data packets.

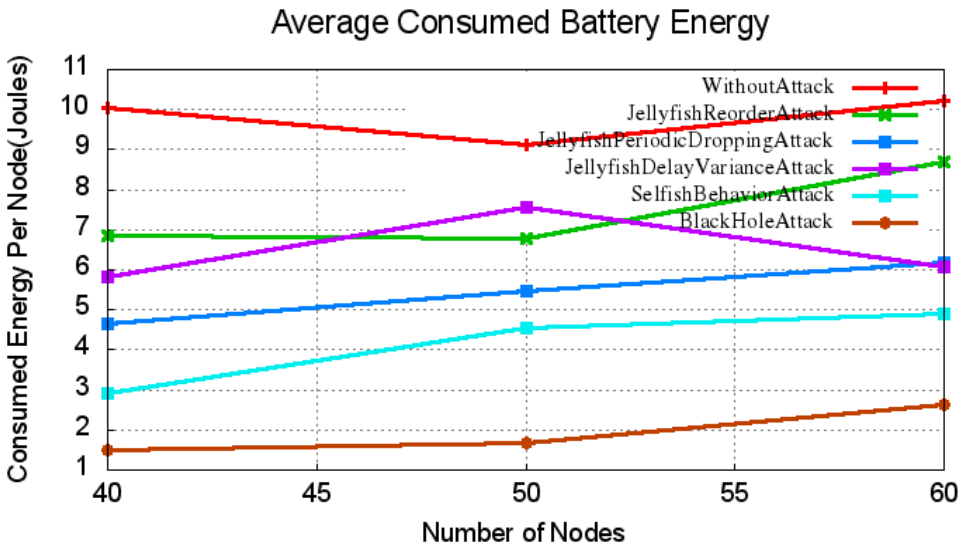

Figure 3.22: Comparison of Network Size Vs Battery Energy on AODV with varying malicious node.

3.3 Summary

In this work, impacts of some of the popular attacks on a short term military rescue mission like MANET scenario has been studied. Pseudo code of attacks and a comparative analysis of three kinds of Jellyfish Attacks and Black Hole Attack with selfish node attack under AODV routing protocol is done [175-177]. Analysis is done with respect to different network sizes and under the presence of different number of attackers in the network. The impact of the attacks with suitable metrics such as sent packet, received packet, dropped packet, throughput, PDF, EED, battery energy etc. has been studied. With respect to the increase of malicious nodes in the network, the performance is getting decreased with respect to the most of the metrics that we considered. Further, with respect to the increase in number of nodes in the network, the performance is getting affected with respect to the nature of attack. Without any doubts, all the attacks affect the performance of AODV routing protocol.

The main scope of this work is to compare the selfish node attack with different Jellyfish Attacks and black hole attack. According to observations and the arrived results, the selfish node attack was as almost worst as black hole attack and even much worse than all types of Jellyfish Attacks with respect to most of the metrics.

Chapter-4

Periodic Trust Handshake Based Malicious Behaviour Detection Mechanisms (PTH-AODV)

As discussed in chapter 2, trust metric is the relationship among the nodes that participate in the routing decisions. It is calculated based on interactions among the nodes. In the current work, trust metric is added to the AODV routing protocol to make it secure and thereby efficient. A new route is chosen based on higher trust value. The trust value is regularly updated in the network based on past communications. The new protocol will overcome the problem and reduce the possibility of false marking of non-malicious nodes as malicious nodes. This protocol is particularly effective in a short time military rescue like MANET scenario [178].

The main advantage of the proposed detection and prevention scheme is: it will detect and prevent the malicious nodes in the very early stage of AODV route discovery process. So, it will not need any manipulation in routing tables in the route resolving process, because, by the design, it will avoid including malicious hops in the routing table even at the route discovery process itself.

Generally, a trust factor based on the packet forwarding behaviour of neighbor can be used for detecting misbehaviour as previously presented in several works in literature. For example, a trust factor of a node can be derived based on the number of forwarded packets at that neighboring node. But, by the same trust based detection logic, some of the

---

The contents of this chapter are under review for publication

[1] Bhawna Singla, A. K. Verma and L. R. Raheja, "Preventing black hole attacks in AODV routing protocols using periodic trust handshake based malicious behaviour detection system" IGI journal of International Journal of Information Security and Privacy (IJISP) (ESCI indexed).

neighbors those who were silent and not actively participated in communications will get low trust factor and will be wrongly identified as malicious. Because of this wrong identification, the link between source to destination will get broken at different locations on their path because of this false identification of malicious nodes. In proposed Periodic Trust Handshake based trust AODV (PTH-AODV), it will overcome that problem and reduce the possibility of such false marking of non-malicious nodes as malicious nodes by introducing a Periodic Trust Handshake mechanism.

4.1 Periodic Trust Handshake Based Malicious Behavior Detection Mechanisms (PTH-AODV)

In this work, trust value is associated with each node and initialized to 0. If the node is working genuinely i.e. forwarding the packet as per the routing protocol instruction then trust value is incremented otherwise it is decremented.

```
IncreaseTrust()
{
        trustValue++;
}
DecreaseTrust()
{
        trustValue--;
}
```

Figure 4.1: Calculation of trust values

Futher, Malicious and selfish nodes are then isolated from the network if they obtain a minimum threshold value.

```
Is_Node_Trusted()
{
        if(trustValue <= threshold value)
        {
                return false;
        }
        else
        {
                return true;
        }
}
```

Figure 4.2: Calculation of malicious node

a) **Packets Acknowledgment**: Acknowledgment helps to ensure that the packet that has been sent for forwarding is indeed forwarded. One common method to implement acknowledgment is the use of passive acknowledgments. Passive acknowledgment means monitor the channel promiscuously. This has been implemented within PTH-AODV.

b) **Packet Precision**: Pirzada et al. [154] defined the packet precision as a method to ensure the integrity of the data and control packets that are either received or forwarded by other nodes in the network. This is achieved by monitoring the control packets that lead to successful route. Another method to achieve this is to check the tolerance limit of packet information. For example, it may be ensured that the sequence number within a reply is not higher than the sequence number within the request.

c) **Destination Unreachable Messages**: Although Pirzada [154] mentions that it is beneficial to use destination unreachable messages but no such messages are used by NS2.

4.1.1 Characteristics of Trust Metric

a) **Storing Trust:** With each node 3 additional classes within AODV are used for storing trust

- TrustNode: is an integer field that stores all the trust information about an individual node
- CircularBuffer: stores information regarding how many packets are there in the cyclic buffer.
- LinkList: the information of all nodes is linked to each other by basic 2 headed linked list.

b) **Promiscuous Mode:** This causes the tap() function within AODV.cc to be called every time a packet is promiscuously seen. Thus, within the tap function, all code related to monitoring other nodes are located.

void PTH-AODV::tap(const Packet *p)

c) **Detecting packet Forwards and Drops:** To detect the successful forwarding of packet, all the sent packets are stored in a cyclic buffer. This buffer is known as a cyclic buffer, defined in the class CircularBuffer. Using a circular buffer means that if packets are not removed frequently enough it will cause the buffer to cycle erasing the last element. This means that if a node is dropping packets or if it is unacceptably slow at forwarding packets then the buffer will start to cycle. Otherwise, if the node is performing acceptably then the packet is added at the end of buffer and increases its trust.

d) **Baring Unwanted Neighbors: In this work, u**nwanted neighbors are bared using by using a call to node_delete(), within the forward() function in AODV.cc. This fuction call will remove the node from the local neighborhood and causes all routes using that node to be removed and a new route request sent out.

```
// If the trust value for the next_hop_ has become too low,
// delete the node from the neighbors


if(!tmpTrustNode->isNodeTrusted(CURRENT_TIME)              ||
    droppedPacket)
{
node_delete(ch->next_hop_);
}
```

Figure 4.3: Removal of unwanted neighbor

4.1.2 The main changes that are made in basic trust based AODV are:

a) Increased Monitoring
b) Weighting Scheme
c) Introduction of a Trust Level
d) Temporary Blacklisting

a) **Increased Monitoring**: All the packets are promiscuously monitored.

b) **Weighting Scheme**: The trust value is set based on various inputs such as direct and indirect observations from the nodes.

```
// Increase the trustValue the amount associated with seeing one of
    the nodes own packets forwarded
void TrustNode::increaseTrust()
{
if( trustValue < MAX_TRUSTVALUE )
{
trustValue++;
}
}
// Decrease the trustValue the amount associated with one of the
    nodes packets not being forwarded timely enough
void TrustNode::decreaseTrust()
{
if( trustValue > -MAX_TRUSTVALUE )
{
trustValue --;
}
}
```

Figure 4.4: Calculation of Trust value

c) **Introduction of a Trust Level**: The trust values are modified between 0 and 1 where 0 means the node is not trusted and 1 means node is completely trusted respectively. Mid value of trust means that we are not sure whether the node is trusted or not.

d) **Temporary Blacklisting**: It also implements a means of temperately baring nodes. The bared node is chosen according to number of packets that have been dropped. This also substantially reduces the packet overhead of the Original PTH-AODV.

4.1.3 Implementation of Periodic Trust Handshake Mechanism

In this model, the nodes will send a "trust handshake" in a periodic fashion. The frequency of this "trust handshake" message will be controlled by a variable max_TrtustHandshake_Interval. This Periodic Trust Handshake mechanism ensures that handshake packet in a periodic fashion so that the neighbor trust factors will be updated

with respect to the mobility of the node. The flowchart 4.6 and the pseudo code in figure 4.7 explain the implementation of Periodic Trust Handshake Based Malicious Node Detection and Prevention in AODV routing agent.

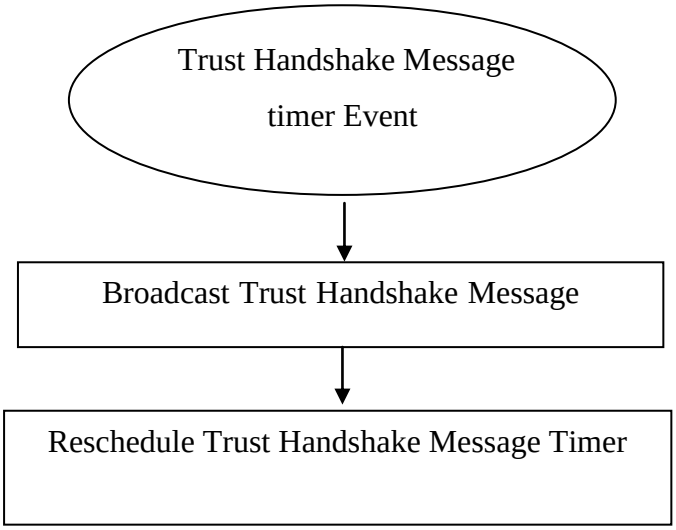

Figure 4.5: The Periodic trust handshake message handler

The process flow of Periodic Trust Handshake Based Malicious Node Detection and Prevention in AODV is shown in figure 4.6. Here, in this algorithm to keep track of selfish node and to distinguish between selfish node and sleeping node (node may be inactive due to no need of transmission of packet), nodes in network keeps on sending the trust metric at periodic time interval. All the nodes also maintain the trust metric of node which is updated on regular time interval. The updation is done based on feedback given by the previous node and nodes along with the best path. The best path is calculated based on distance and trust metric. The following two files were modified to incorporate the proposed malicious node detection and prevention mechanism in AODV routing agent.

a) Changes made in AODV.h: The additional function definitions for detection and prevention of malicious behavior and the variables that will be bound with TCL are declared in AODV.h.

b) Changes made in AODV.cc: The actual code of the additional function definitions for detection and prevention of malicious behavior were implemented in AODV.cc. And here the new interfaces to the code through the control variables that will be bound with TCL are written here. By setting the variables from a TCL simulation

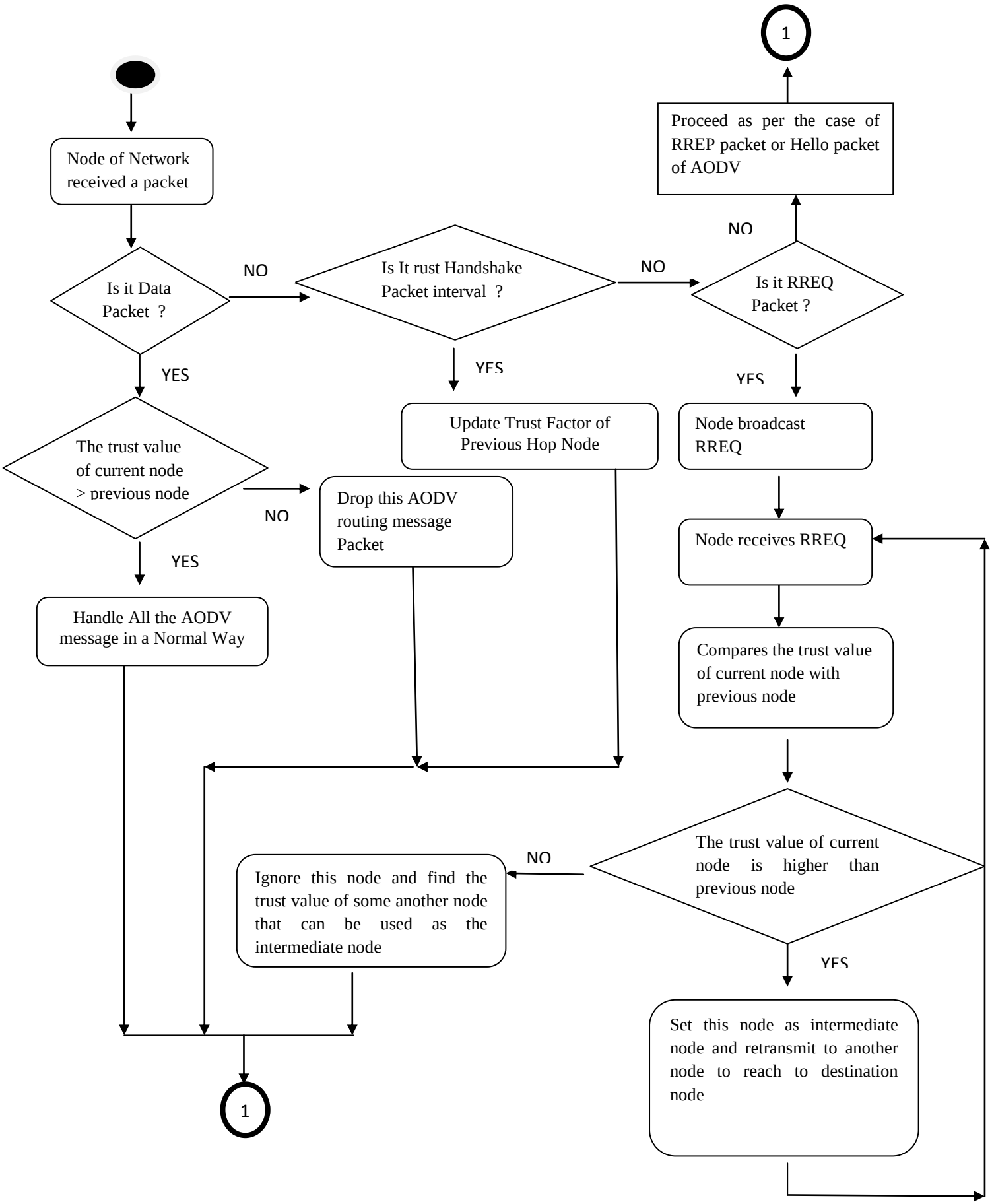

Figure 4.6: The process flow of Periodic Trust Handshake Based Malicious Node Detection and Prevention in AODV.

code, we can control the behavior of the routing agent and bring it to detection and prevention mode.

The following functions were modified to incorporate the proposed malicious node detection and prevention mechanism in AODV routing agent.

a) The function TrustHandshakeTimer(): The Periodic Trust Handshake Mechanism is implemented with the help of a new timer function in AODV.

b) The function AODV::SendTrustHandshakePacket() This function will generate a Trust Handshake packet and transmit it with respect to the conditions explained in the figure 4.5.

c) The function AODV::recvAODV() : In this function, the trust based detection of malicious behavior has been implemented. As shown in the figure 4.10, the malicious behavior detection is done based on the trust factor of the previous hop node from which the message was received.

```
Forward(RREQ pkt, delay) {

// the node receives the RREQ control packet
// checks whether it is destination node
if destination{
        //considers the path with highest trust value and sends the route reply along that
        path
        compute_highest_trust_level ()
                {
                // the optimal path with highest value value is chosen and route reply is
                sent along that path
                highest_trust_value(path)
                sends_RREP_to_source
                }
else (not_destination){
        // if the next intermediate node is not destination then intermediate node checks
        for the packet by computing the trust level
        if RREQ_packet{
        compute_trust_level ()
```

```
                {
                // compares the trust value of current node with the tryst value of previous
                node
                trust_current_node> trust_previous_node
                }
        if (found not ok)
        // intermediate node drops the packet if its trust level is lesser then previous path
        drop(pkt)
        else  {
        // if the new path has more trust value then update trust and hop count and
        rebroadcast it to next neighbour node
         trust++
        // total number of intermediate nodes is incremented by 1
        hop_count++

        // the RREQ packet is rebroadcasted to next neighbour node
        rebroadcast RREQ
                        }
                }
        }
        }

Receive (RREP pkt, delay) {
        //waits for specified period
        if no_duplicate{
                wait_rrep_wait_time
                update_trust_metric
                update next_hop}
                else
                {
                compute trust_path
                }
        }
```

```
Update_Trust_Metric (interval){
  //wait for the minimum trust handshake interval
  wait_trust_handshake_interval();
  broadcast_trust_hanshake;
  trust_value>trust_threshold {
        trust_current_node = trust_current_node + trust_previous_node
              }
        else {
              drop (pkt);}
        }
```

Figure 4.7: The Pseudo code of PTH-AODV

4.2 Simulation Parameters

a) Common Parameters: The following common parameters are used for setting up the network. Moreover, following parameters are also used to set TCP/UDP flows.

Table 4.1: Parameters values of Network in NS2

Common parameters	**Values**	**Traffic parameters for TCP flows**	**Values**
Topographical Area (m*m)	1800 X 500	Transport Agent	TCP
Mobility	20m/s	No Flows	10
Pause Time	20s	Traffic Type	CBR
Total SimulationTime	100s	Packet Size	1KB
Routing Protocol	AODV	Interval	100ms
Mobility Modal	RandomWaypoint	Rate	10KB
Channel Model	WirelessChannel	Traffic parameters for UDP flows	Values
Propagation Model	TwoRayGround	Transport Agent	UDP
PhyModel	WirelessPhy	No Flows	10
MacModel	802_11	Traffic Type	CBR

AntennaModel	OmniAntenna	Packet Size	1KB
Queue	DropTail-PriQueue	Interval	100ms
Queue Length	50	Rate	10KB

b) Variable Parameters: The following parameters are used as variables for analyzing the impact of the attack and detection on different condition.

Table 4.2: Total number of nodes, number of malicious node and different attack scenarios

Parameters	Values
Malicious Nodes	15
Total Nodes	40,50,60
AODV with	a) No Attack b) Black Hole Attack c) PTH Attack Detection

4.3 Results of PTH-AODV

4.3.1 Analysis of Results with respect to different network size

Here we see the analytic results of comparison of black hole attacks with AODV (it means performance without any attack). And it is studied with respect to different network size. In the following analysis the total number of nodes in the network is varied as 40, 50 and 60 and among them, the number of malicious nodes kept as 15 and the impact is measured using different metrics.

The graph in figure 4.8 shows the impact of attack and detection and prevention mechanism in terms of total data packets sent at application source. As shown in the graph 4.8, under the presence of black hole attack the application source itself can not able to send much. But while detection the proposed PTH-AODV was able to send as much as AODV sends without any attack.

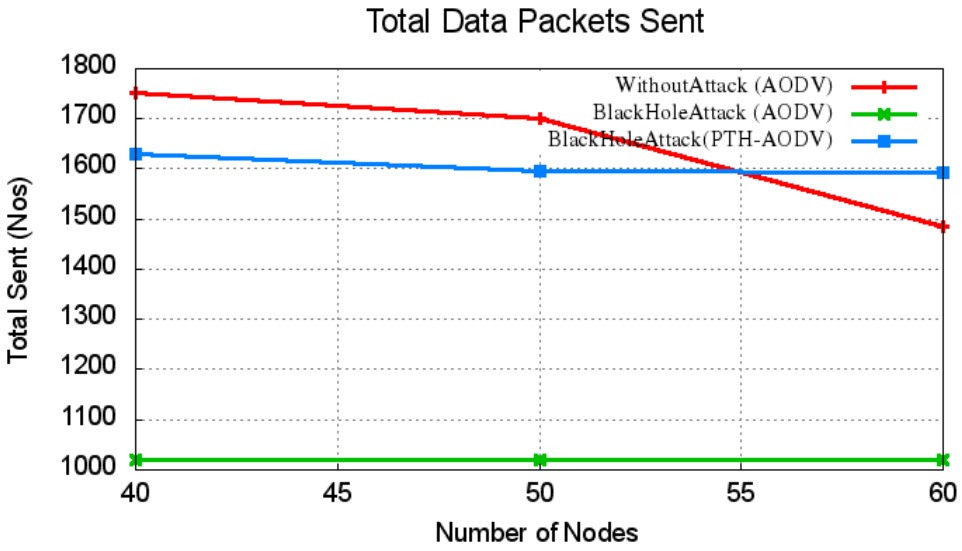

Figure 4.8: Comparison of Network Size Vs Sent Packets in PTH-AODV.

The graph in figure 4.9 shows the impact of attack and detection and prevention mechanism in terms of total data packets received at application destination. As shown in the line graph, under the presence of black hole attack the application destination itself can not able to receive anything. But while detection the proposed PTH-AODV was able to receive as much as AODV without any attack.

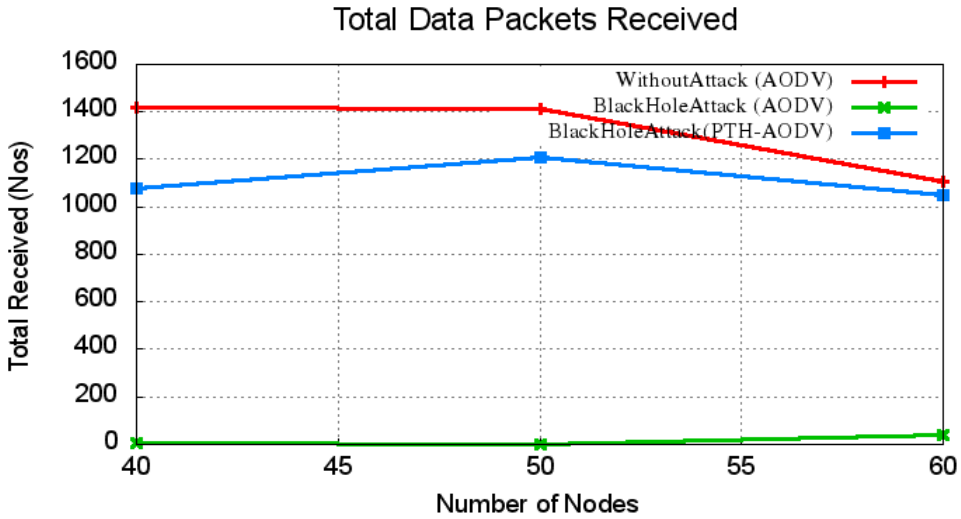

Figure 4.9: Comparison of Network Size Vs Received Packets in PTH-AODV.

The graph in figure 4.10 shows the impact of attack and detection and prevention mechanism in terms of routing load. As shown in the line graph, under the presence of Black hole the routing load is very high. But with proposed PTH-AODV based detection and prevention mechanism, the routing load was almost equal to that of AODV.

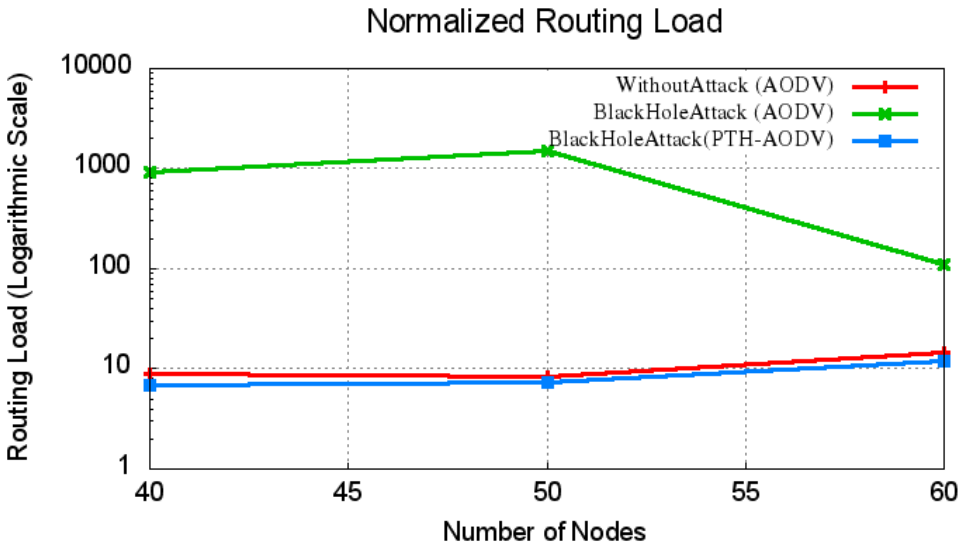

Figure 4.10: Comparison of Network Size Vs Routing Load in PTH-AODV.

The graph in figure 4.11 shows the impact of attack and detection and prevention mechanism in terms of MAC load. As shown in the line graph, under the presence of Black hole the MAC load is very high. But with proposed PTH-AODV based detection and prevention mechanism, the MAC load was almost equal to that of AODV.

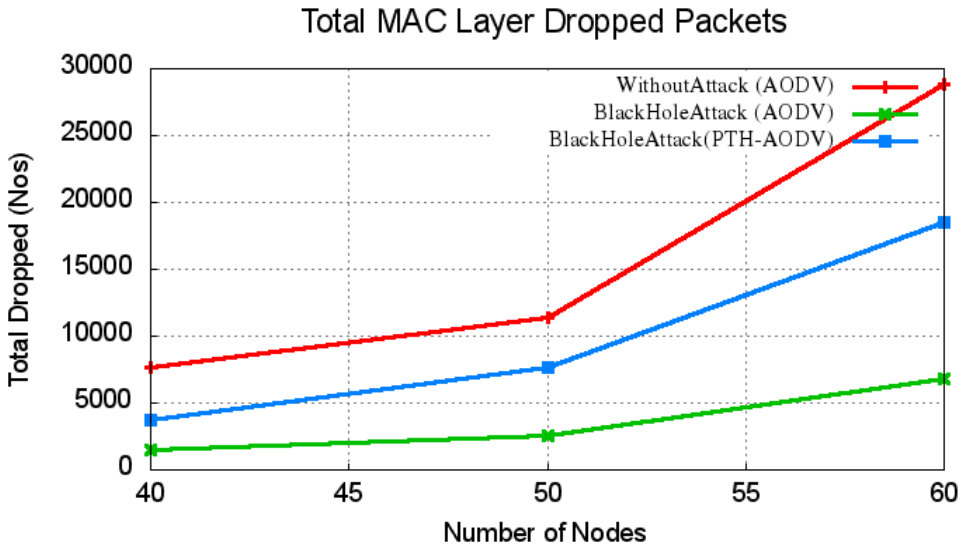

Figure 4.11: Comparison of Network Size Vs MAC Load in PTH-AODV.

The graph in figure 4.12 shows the impact of attack and detection and prevention mechanism in terms of total dropped packets at the application layer. As shown in the line graph, under the presence of black hole attack a lot of packets were dropped at the application layer. But while detection, the packet dropping of proposed PTH-AODV was very much reduced and almost equal to that of AODV without any attack.

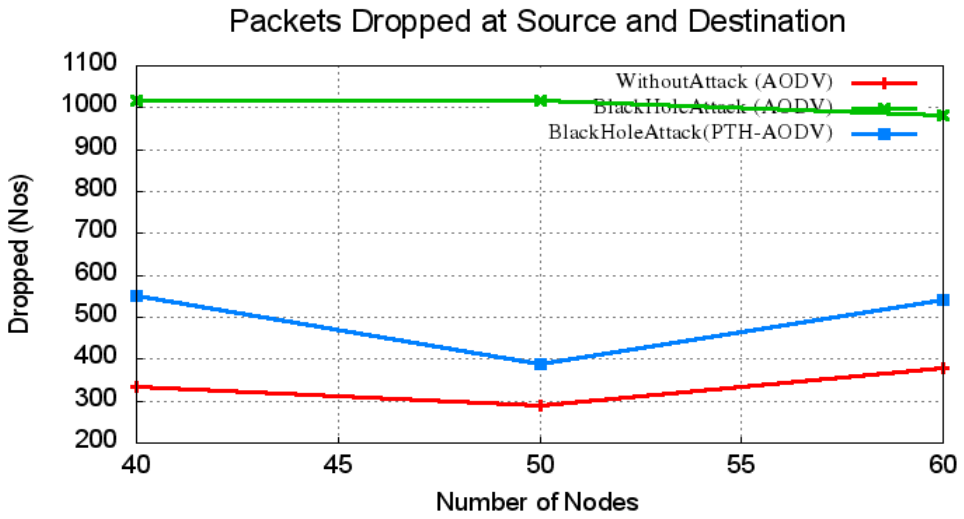

Figure 4.12: Comparison of Network Size Vs Packets Dropped At The application layer in PTH-AODV.

The graph in figure 4.13 the impact of attack and detection and prevention mechanism in terms of throughput. As shown in the line graph, under the presence of black hole attack the throughput was almost equal to zero. But with detection, the throughput of proposed PTH-AODV was very much improved and almost equal to that of AODV without any attack. This figure also shows the variation of results when it is re-executed over 10 simulations. This simulation is repeated so as to verify the result of PTH-AODV. It is observed that the PTH-AODV gives the result within the range of ±3%. It shows that proposed PTH-AODV results may vary within the range of ±3%. This fact can also be utilized to compare the result of PTH-AODV with the other existing secure routing protocol.

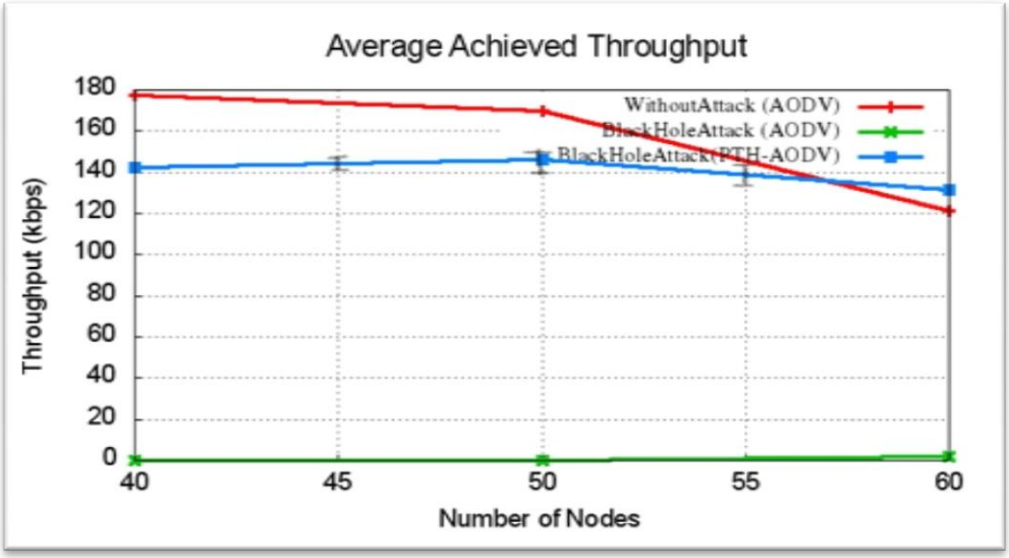

Figure 4.13: Comparison of Network Size Vs Throughput in PTH-AODV.

The graph in figure 4.14 shows the impact of attack and detection and prevention mechanism in terms of PDF. As shown in the line graph, under the presence of black hole attack the PDF was almost equal to zero. And at low network density PDF is equal to zero. For example, at 40 nodes, it is zero because, among the 40 nodes, 15 are malicious-so that they will able to break all the communication between other nodes. But with detection, the PDF of proposed PTH-AODV was very much improved and almost equal to that of AODV without any attack.

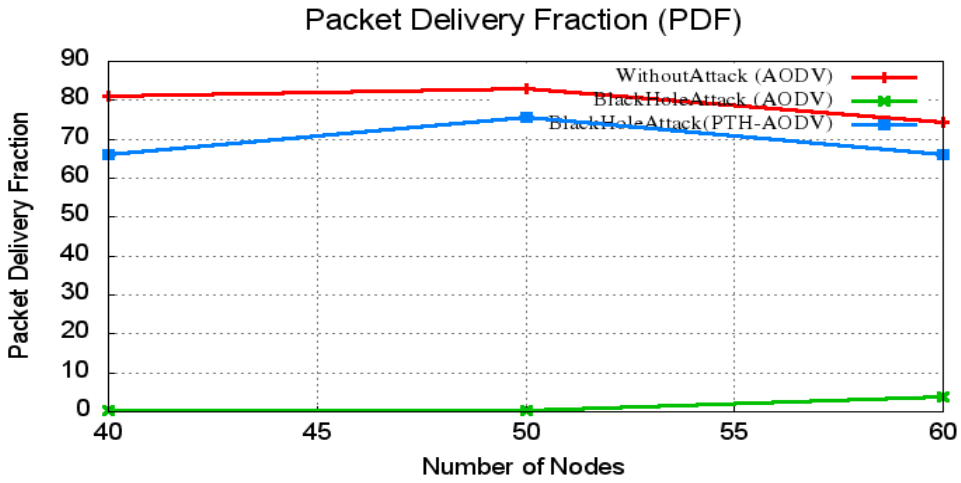

Figure 4.14: Comparison of Network Size Vs PDF in PTH-AODV.

The graph in figure 4.15 shows the impact of attack and detection and prevention mechanism in terms of End-to-end Delay (EED) of data flows. With respect to the increase of number of nodes in the network, the performance getting decreased .As shown in the line graph, black hole attack seems to be providing lower EED than AODV(without attack) – but certainly, it does not mean that black hole attack is improving the performance of the network. The low EED under attack is due to a strange fact that the attack makes disconnection in TCP flows and since the packets are not at all forwarded to any further nodes, indirectly it is reduce the message overhead in the network and reduced bandwidth usage otherwise it will be consumed by the data packets. So, the flows that were unaffected by black hole attack (the connections where there is no neighboring attack nodes) utilizes that extra bandwidth and gains some

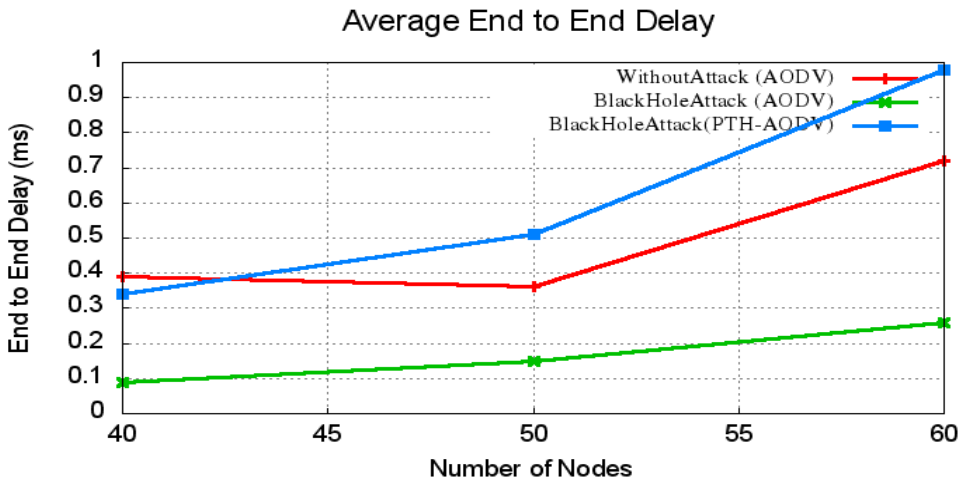

Figure 4.15: Comparison of Network Size Vs End-to-end Delay in PTH-AODV.

performance in term of some metrics.

The EED of PTH-AODV was a little bit higher than AODV. Because, under attack detection and prevention, alternate route will be resolved by avoiding malicious nodes on a path, So that the path length will get increased and hence will increase the EED.

The graph in figure 4.16 shows the impact of attack and detection and prevention mechanism in terms of consumed battery energy. As shown in the line graph, in the presence of Attack the battery consumption is lesser than AODV (without attack) – but certainly it does not mean these Attacks are improving the performance in terms of energy consumption. The low energy consumption under attacks are due to a strange fact that these attacks makes disconnection in data flows and since the packets are not at all

forwarded to any further nodes, indirectly it is reduce the battery consumption at the other nodes otherwise it will be consumed for forwarding the data packets. So, the nodes that were unaffected by Attacks (where there is no neighboring attack nodes) preserves some battery power. Understanding this strange fact requires a better visualization of the whole network scenario. It is simple – without any attack, AODV was able to send much and maximum nodes were able to participate in that communication and utilized their energy for transmission/forwarding of packets – so that the energy is consumed in most of the nodes. But in the presence of attack, the packets are getting dropped intermediately and the battery powers on other nodes that are not at all forwarding the packets gets preserved. With respect to the increase of no of nodes in the network, the performance seems to be getting decreasing.

But, interestingly, the energy consumption in the case of proposed PTH-AODV is a little bit lesser than AODV. This obviously proves the better working of proposed detection model.

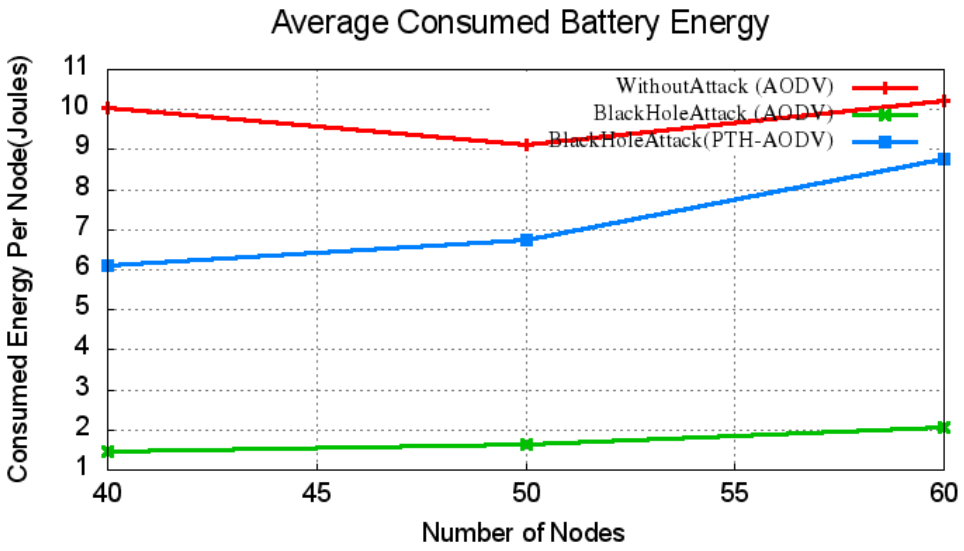

Figure 4.16: Comparison of Network Size Vs Battery Energy in PTH-AODV.

The graph in figure 4.17 shows the impact of attack and detection and prevention mechanism in terms of overhead. As shown in the line graph, under the presence of Black hole the overhead is minimum – because, the black holes just break all the communication. But with proposed PTH-AODV based detection and prevention mechanism, the overhead becomes equal to that of AODV – it signifies that the proposed PTH-AODV works almost equal to AODV.

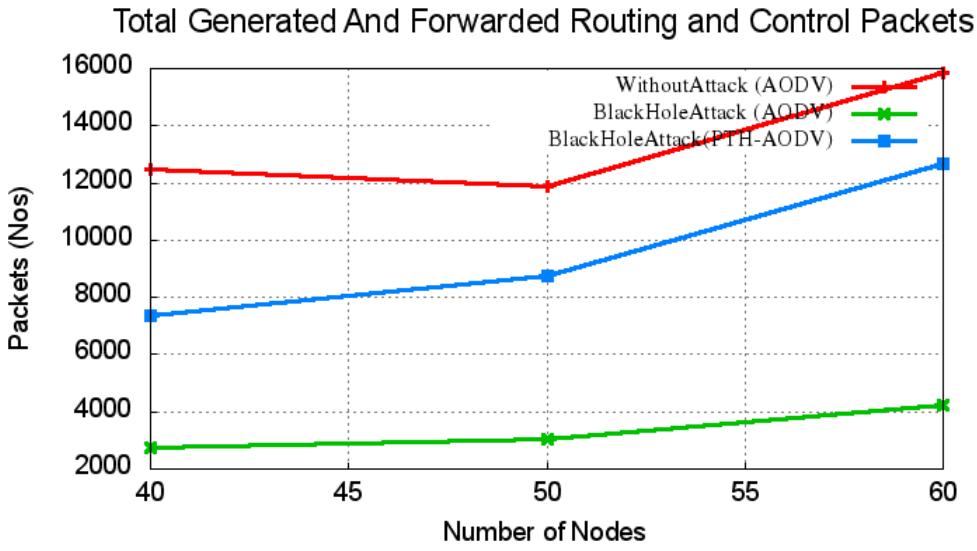

Figure 4.17: Comparison of Network Size Vs Overhead in PTH-AODV.

The graph in figure 4.18 shows the impact of attack and detection and prevention mechanism in terms of total maliciously dropped packets at MAC layer. For the first look, one may think as this as a wrong result because of the decrease in malicious dropping in the case of attack as well as detection and prevention (PTH-AODV). But it is not. The dropping in the case of black hole attack is decreased because, the malicious packet dropping is only happening at routing layer. The dropping in the case of attack is less than all because, PTH-AODV a little bit higher than attack without detection because, PTH-AODV will try to avoid black holes so that, initiate new route discovery process and this causes more packet generation and loss at MAC layer.

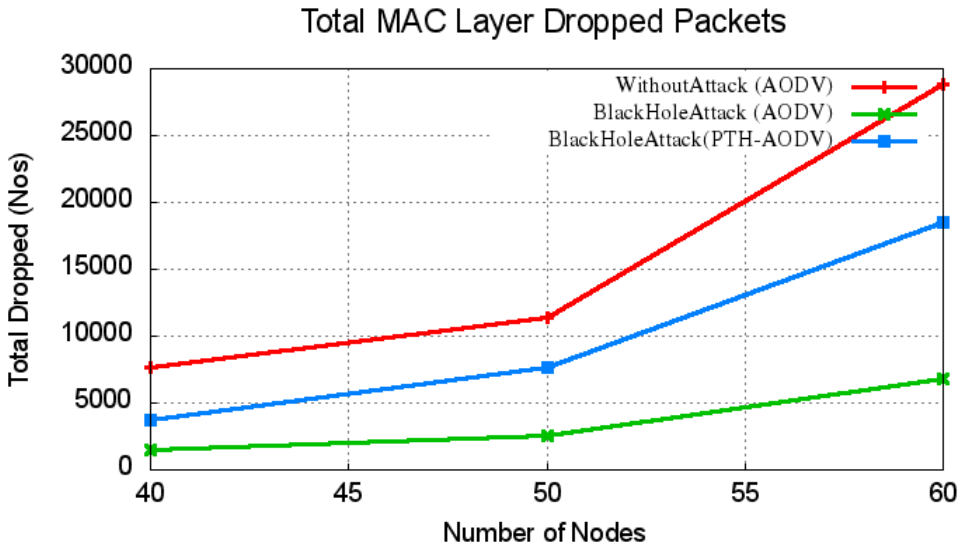

Figure 4.18: Comparison of Network Size Vs MAC Layer Dropped in PTH-AODV.

The graph in figure 4.19 shows the impact of attack and detection and prevention mechanism in terms of total maliciously dropped packets at network layer. For the first look, one may think as this as a wrong result because of the increase in malicious

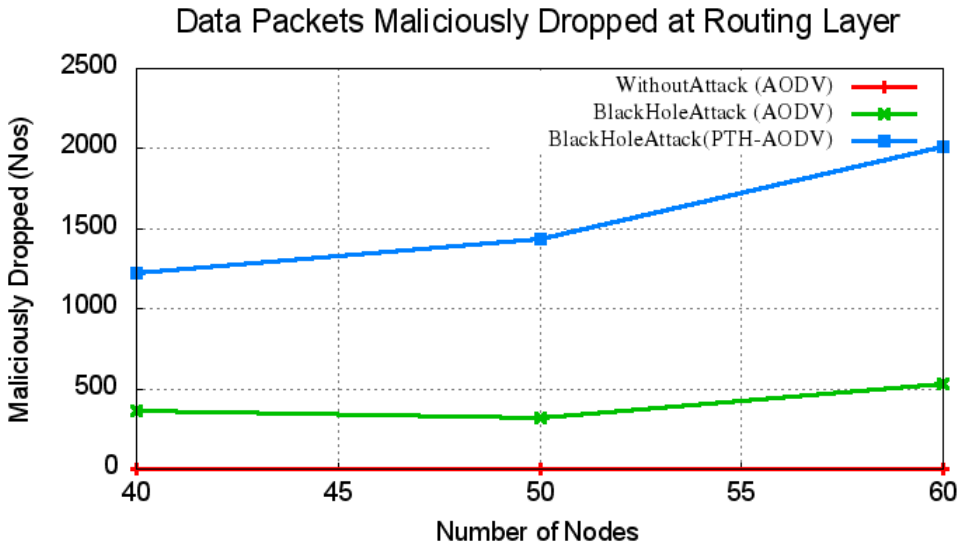

Figure 4.19: Comparison of Network Size Vs Malicious Drops at Routing Layer in PTH-AODV.

dropping in the case of detection and prevention (PTH-AODV). But it is not. The malicious dropping in the case of PTH-AODV is increase because it is trying to send the packet in one way or another by avoiding malicious nodes. The retransmissions involved in this process increases malicious packet dropping.

4.4 Comparison of PTH-AODV with existing secure routing protocols

Test cases of PTH-AODV protocol with the protocol discussed in "Trust-Based Routing Mechanism in MANET: Design and Implementation" [159] is compared so as to determine the efficiency of PTH-AODV. And assumptions taken are as: The number of mobile nodes is taken as 100, simulation time is taken as 250s, pause time is varied from 0 to 250s in a environment of 1000 m*1000 m. The following tests were conducted to compare the performance of protocols.

Test: Varying the node pause time: Figure 4.20 shows the performance results of fr-AODV [159] and PTH-AODV by varying the pause time. Pause time indicates the movement of nodes. Pause time equal to 0 shows highly mobile environment and higher

pause time indicates the fixed network. It is observed in the figure that moderate pause time results better in case of PTH-AODV.

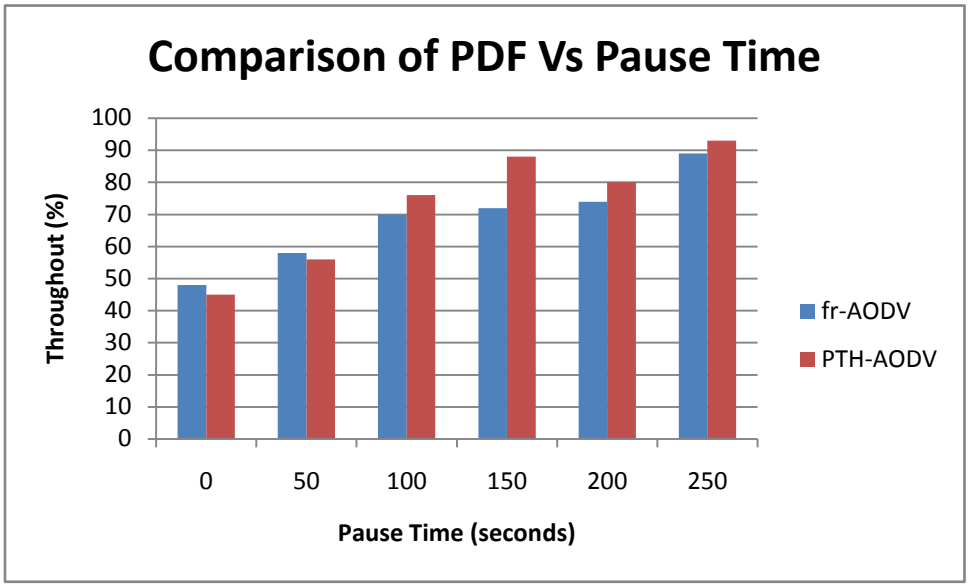

Figure 4.20: Comparison of various routing protocol by varying the Pause time

4.5 Summary

In this work, periodic trust handshake based detection of black hole attack is proposed. PTH-AODV under NS2 is implemented and compared its performance with the results of AODV and AODV under attack. The main advantage of the proposed PTH-AODV is : it will detect and prevent the malicious nodes in the very early stage of route discovery process. So, it will not need any manipulation in routing tables in the route resolving process, because, by the design, it will avoid including malicious hops in routing table of normal nodes at the route discovery process itself.

The impact of the attack as well as the detection and prevention mechanism with suitable metrics such as sent packet, received packer, dropped packet, throughput, PDF, EED, energy is observed. According to the arrived results, proposed periodic trust handshake based malicious node detection and prevention mechanism worked good and successfully detected black hole nodes in the network and avoided establishing routes through them. As shown in the results, the proposed PTH-AODV improved the throughput and PDF almost equal to that of AODV. The results of PTH-AODV are also verified by comparing it with T-AODV and PTH-AODV was found to perform better.

Chapter-5

Dynamic Trust Handshake Based Malicious Behaviour Detection Mechanisms (DTH-AODV)

To avoid false positives, and to improve the detection accuracy, the use of a Periodic Trust Handshake mechanism is proposed. In this chapter, the performance of the algorithm using a Dynamic Trust Handshake based detection mechanism is further increased. Dynamic Trust Handshake based detection mechanism will detect the malicious nodes very quickly and efficiently in a short time military rescue like MANET scenario without much increase in overhead.

The contents of this chapter have been peer reviewed and accepted for publication

[1] Bhawna Singla, A. K. Verma and L. R. Raheja, "Preventing Black hole Attack in AODV Routing Protocol using Dynamic Trust Handshake Based Malicious Behaviour Detection Mechanism" Machine Learning for Computer and Cyber Security: Principles, Algorithms, and Practices (acceptance 21-04-2018).

5.1 Dynamic Trust Handshake Based Malicious Behaviour Detection Mechanism

One of the critical problems in MANETs is the security vulnerabilities of the routing protocols. Node misbehavior due to malicious intention could significantly degrade the performance of MANET because the most existing routing protocols in MANET are aiming at finding most optimal path.

Detection of black hole is a challenging task. Further, isolating such malicious nodes from communication is also a great challenge. Several previous works address trust based model for detection and prevention of malicious nodes. Trust based models will consume time to study the neighbor transmissions and will try to identify trustable nodes based on their data forwarding behavior. But this approach will need a considerable quantity of time to identify malicious nodes by constantly monitoring the traffic of the neighbor nodes. Another drawback in this model is, false positives – that is, the standard trust based detection mechanisms may wrongly mark a trustable node as non-trustable node if that node, by chance, is not participating in communication even without any bad intention. To prove its better working, we simulated a MANET short time communication scenario and measured the performance of AODV with and without black hole attack and compared it with Dynamic Packet Forwarding based Trust AODV (DTH-AODV) protocol in terms of different metrics. The proposed DTH-AODV will use a Dynamic Trust Handshake mechanism for the reliable detection of malicious behavior in MANET,

5.1.1 Implementation of Dynamic Trust Handshake Mechanism

Generally, a trust factor based on the packet forwarding behavior of neighbor can be used for detecting misbehavior as previously presented in several works in literature. For example, a trust factor of a node can be derived based on the number of forwarded packets at that neighboring node. But, by the same trust based detection logic, some of the neighbors those who were silent and not actively participated in communications will get low trust factor and will be wrongly identified as malicious. Because of this, the link between source nodes to destination will get broken at different locations on their path because of this false identification of malicious nodes. In the proposed dynamic trust handshake based AODV (DTH-AODV), it will overcome that problem and reduce the possibility of such false marking of non-malicious nodes as malicious nodes by

introducing a Dynamic Trust Handshake mechanism. The following flowchart explains the implementation of Dynamic Trust Handshake Mechanism in AODV routing agent.

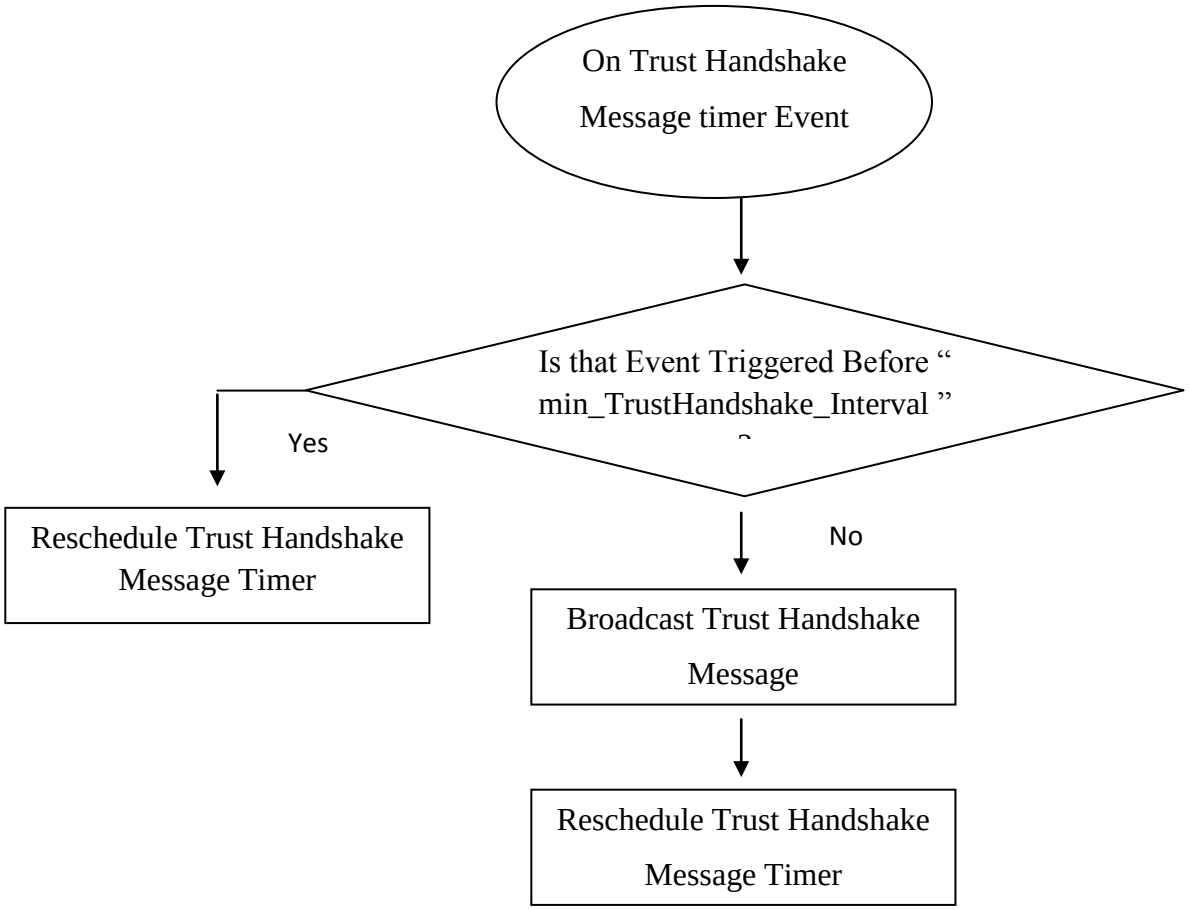

Figure 5.1: Dynamic Trust handshake mechanism

In previously implemented PTH-AODV, trust handshake message will be sent only in periodic intervals but in this proposed DTH-AODV, that was dynamically controlled with respect to the state of the routing process.

In this model, the nodes will sent a "trust handshake" in a dynamic fashion based on its local state. This Dynamic Trust Handshake mechanism ensures that at least one handshake packet will be send just before any new transmission event. But the frequency of such "trust handshake" message will be controlled by two variables the min_TrustHandshake_Interval and max_TrustHandshake_Interval. So, it will not increase the message overhead tremendously.

The trust handshake message function will be called from a different function of AODV whenever a change in state is expected. For example, after doing a regular route table update, the trust handshake message function will be triggered. But according to the way in which The Dynamic Trust Handshake Mechanism working, it will not actually send a handshake message whenever it is triggered. The trigger mechanism may rapidly call the trust handshake message sending function, but it will actually send a new message if an only if there was a considerable gap (min_TrustHandshake_Interval) between two consecutive messages. This will avoid over sending the Trust Handshake messages. The following flowchart explains the implementation of Dynamic Trust Handshake Based Malicious Node Detection and Prevention in AODV routing agent.

The following two files were modified to incorporate the proposed malicious node detection and prevention mechanism in AODV routing agent.

a) Changes Made in AODV.h: The additional function definitions for detection and prevention of malicious behavior and the variables that will be bound with TCL are declared in AODV.h. By using the variables from a TCL simulation code, we can control the behavior of the routing agent and bring it to detection and prevention mode.

b) Changes Made in AODV.cc: The actual code of the additional function definitions for detection and prevention of malicious behavior was implemented in AODV.cc. And here the new interfaces to the code through the control variables that will be bound with TCL are written here. By setting the variables from a TCL simulation code, we can control the behavior of the routing agent and bring it to detection and prevention mode.

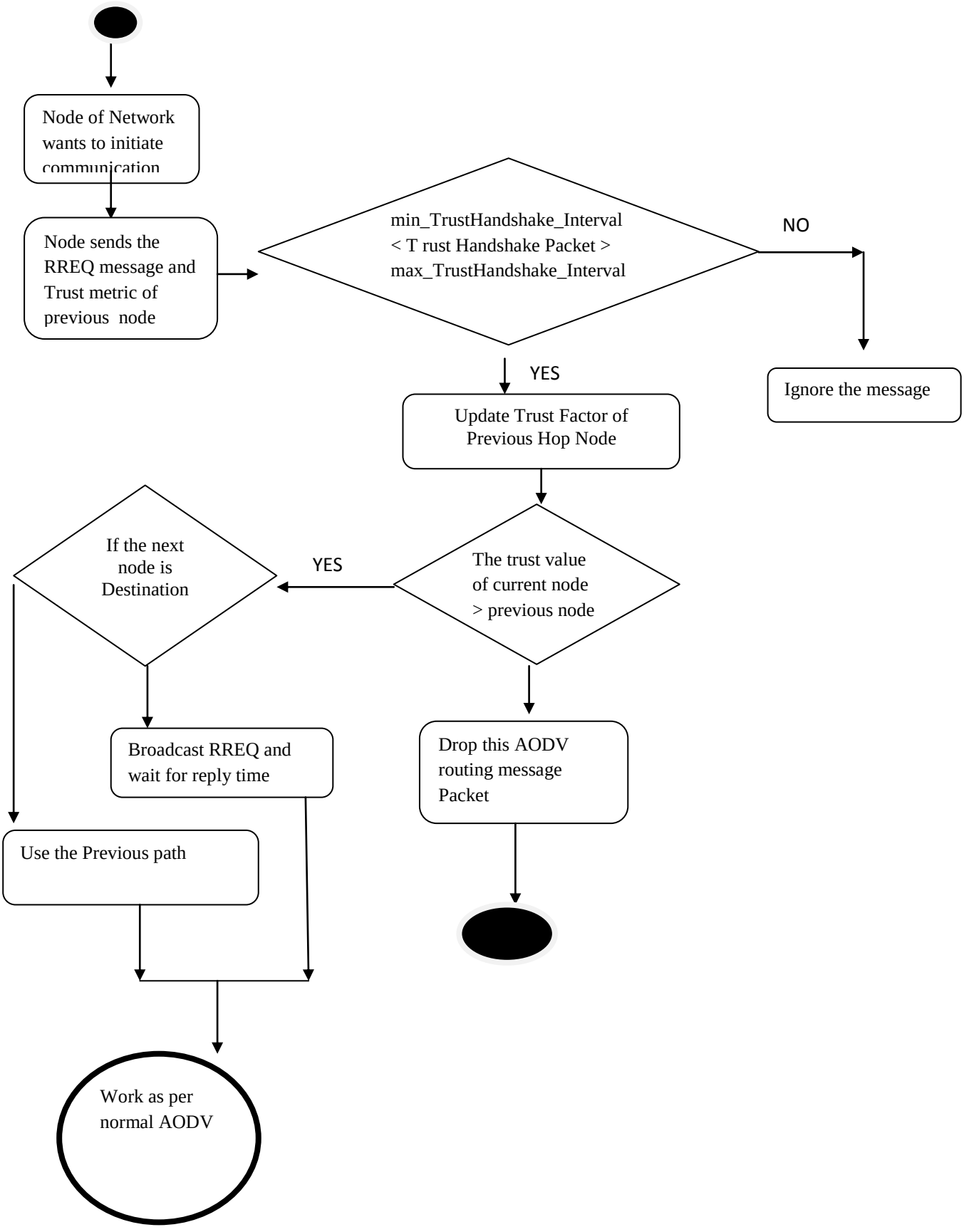

Figure 5.2: Process Flow of Dynamic Trust handshake mechanism

```
Forward(RREQ pkt, delay) {
    // the node receives the RREQ control packet
    // checks whether it is destination node
        if destination{
        //considers the path with highest trust value and sends the route reply along
        that path
        compute_highest_trust_level ()
            {
            // the optimal path with highest value value is chosen and route
            reply is sent along that path
            highest_trust_value(path)
            sends_RREP_to_source
            }
    else (not_destination){
        Update_trust_value{
        //Exchanges the trust field of various nodes on the bases of direct feedback
        to update the trust value
        Trust_exchange() }
        // if the next intermediate node is not destination then  intermediate node
        checks for the packet  by computing the trust level
        if RREQ_packet{
        compute_trust_level ()
            {
            // compares the trust value of current node with the tryst value of
            previous node
            trust_current_node> trust_previous_node
            }
        if (found not ok)
        // intermediate node drops the packet if its trust level is lesser then
        previous path
        drop(pkt)
        else  {
```

```
// if the new path has more trust value then update trust and hop count and
rebroadcast it to next neighbour node
 trust++
// total number of intermediate nodes is incremented by 1
hop_count++
// the RREQ packet is rebroadcasted to next neighbour node
rebroadcast RREQ
                }
                }
        }
        }

    Receive (RREP pkt, delay) {
        //waits for specified period
        if no_duplicate{
            wait_rrep_wait_time
            update_trust_metric
            update next_hop}
            else
            {
            compute trust_path
            }
            }
        }

Send data(packet,trust) {
        Update_trust_value{
        //Exchanges the trust field of various nodes on the bases of direct feedback
        to update the trust value
        Trust_exchange()
             }
        }
```

Figure 5.3: The pseudo code of DTH-AODV

The following functions were also added to incorporate malicious node detection and prevention mechanism.

a) The function TrustHandshakeTimer(): The Periodic Trust Handshake Mechanism is implemented with the help of a new timer function in AODV.

b) The function AODV::SendTrustHandshakePacket(): This function will generate a Trust Handshake packet and transmit it with respect to the conditions explained in the figure 5.1.

c) The function AODV::recvAODV(): In this function, the trust based detection of malicious behavior has been implemented. As shown in the figure () of previous section, figure 5.1, the malicious behavior detection is done based on the trust factor of the previous hop node from which the message was received.

5.2 Results of DTH-AODV

We used network simulator version NS2.35 under Ubuntu linux operating system for obtaining this results. We have implemented the black hole attack as well as attack detection and prevention mechanism on the AODV code of NS2 and did the simulation with the parameters presented in this section and evaluated the performance with respect to the metrics discussed in this section.

5.2.1 Analysis of Results with respect to different network size

Here analytic results of comparison of black hole attacks with AODV (it means performance without any attack) is observed. And it is studied with Respect to Different Network Size. In the following analysis the total number of nodes in the network is varied as 40, 50 and 60 and among them, the number of malicious nodes kept as 15 and the impact is measured using different metrics.

The graph in figure 5.4 shows the impact of attack and detection and prevention mechanism in terms of total data packets sent at application source. As shown in the line graph, under the presence of black hole Attack the application source itself can not able to send much. But while detection the proposed DTH-AODV was able to send as much as AODV without any attack. In terms of send packets, the proposed DTH-AODV performed a little bit better than previous PTH-AODV.

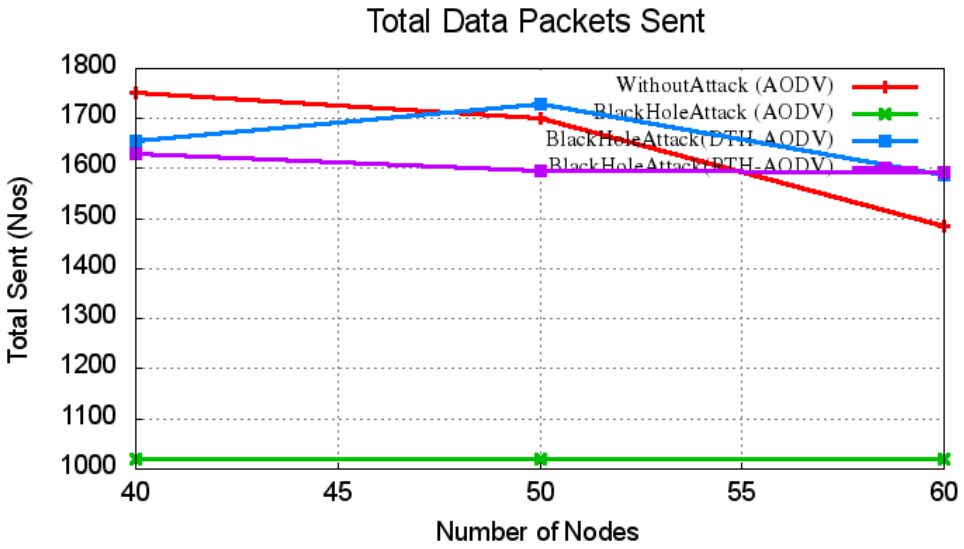

Figure 5.4: Comparison of Network Size Vs Sent Packets in DTH-AODV.

The graph in figure 5.5 shows the impact of attack and detection and prevention mechanism in terms of total data packets received at application destination. As shown in the line graph, under the presence of black hole Attack the application destination itself can not able to receive anything. But while detection the proposed DTH-AODV was able to receive as much as AODV without any attack. In terms of received packets, the proposed DTH-AODV performed a little bit better than previous PTH-AODV.

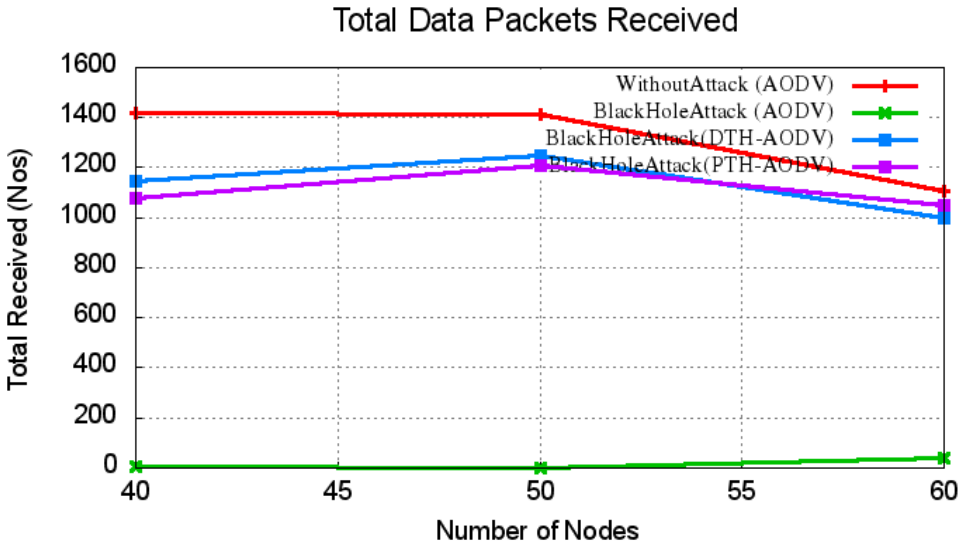

Figure 5.5: Comparison of Network Size Vs Received Packets in DTH-AODV.

The graph in figure 5.6 shows the impact of attack and detection and prevention mechanism in terms of routing load. As shown in the line graph, under the presence of black hole the routing load is very high. But with proposed DTH-AODV based detection and prevention mechanism, the routing load was almost equal to that of AODV. In terms of routing load, the performance of AODV, proposed DTH-AODV and the previous PTH-AODV are almost equal.

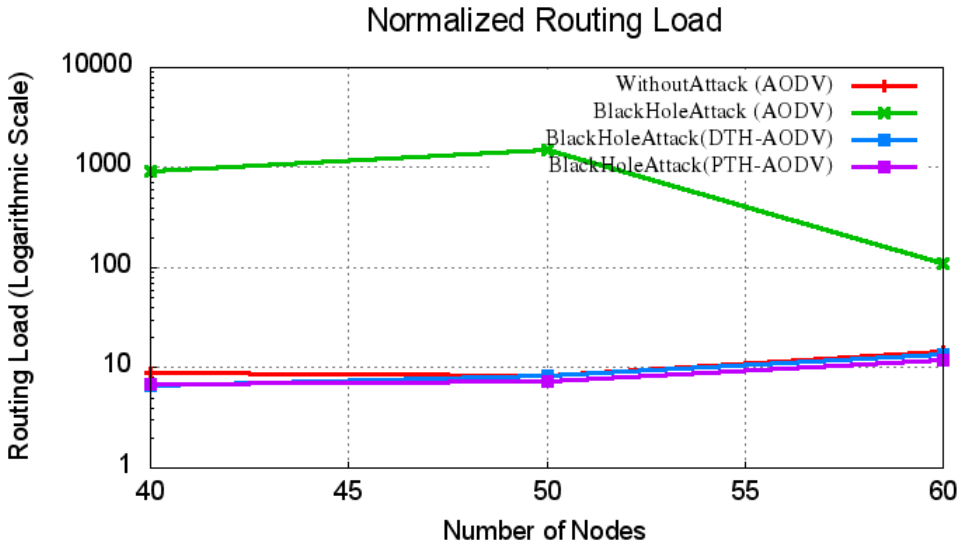

Figure 5.6: Comparison of Network Size Vs Routing Load in DTH-AODV.

The graph in figure 5.7 shows the impact of attack and detection and prevention mechanism in terms of MAC load. As shown in the line graph, under the presence of black hole the MAC load is very high. But with proposed DTH-AODV based detection and prevention mechanism, the MAC load was almost equal to that of AODV. In terms of MAC load, the performance of AODV, proposed DTH-AODV and previous PTH-AODV are almost equal.

The graph in figure 5.8 shows the impact of attack and detection and prevention mechanism in terms of total dropped packets at the application layer. As shown in the line graph, under the presence of black hole attack a lot of packets were dropped at the application layer. But while detection, the packet dropping of proposed DTH-AODV was very much reduced and almost equal to that of AODV without any attack. In terms of the application layer dropped packets, the proposed DTH-AODV dropped a little bit high number of packets than the previous PTH-AODV – this is because, the DTH-AODV will try to send more packets than AODV and PTH-AODV.

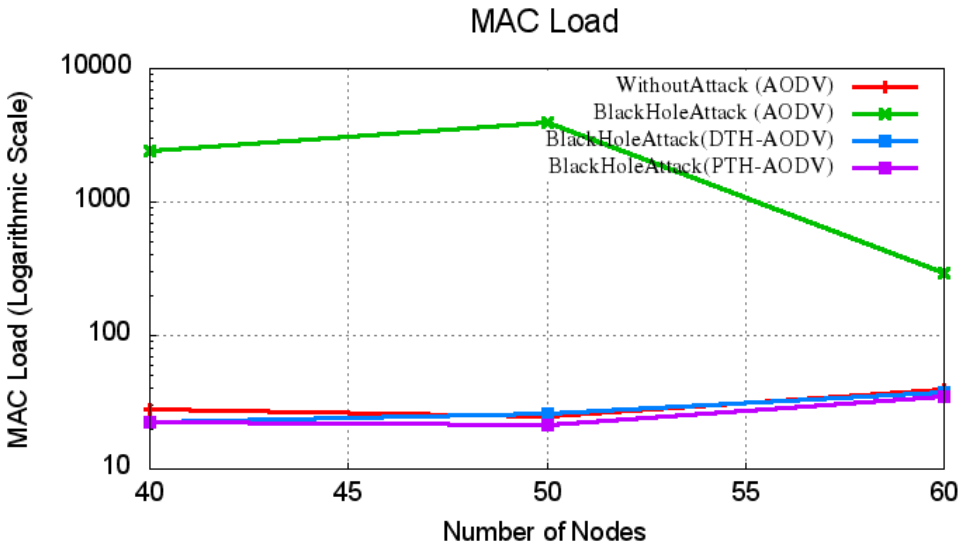

Figure 5.7: Comparison of Network Size Vs MAC Load in DTH-AODV.

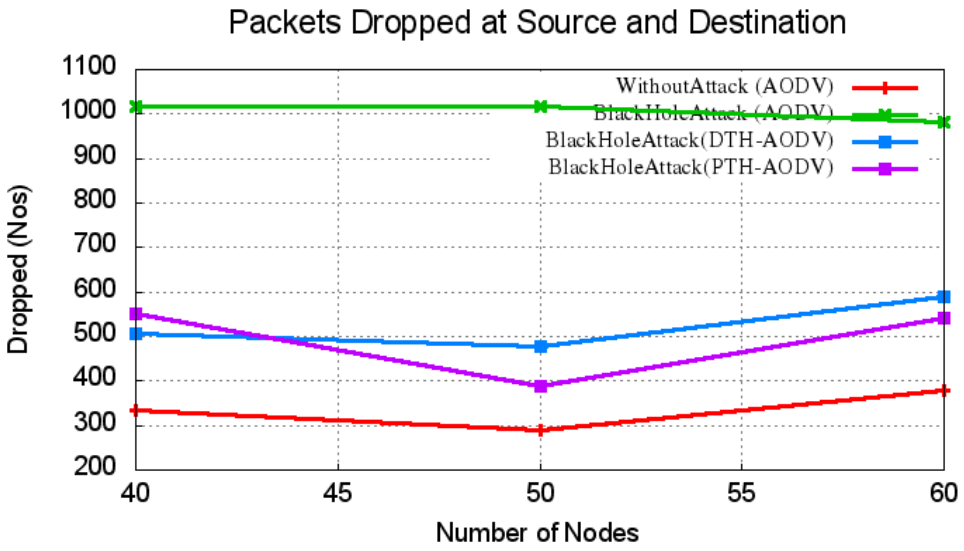

Figure 5.8: Comparison of Network Size Vs Packets Dropped At The application layer in DTH-AODV.

The graph in figure 5.9 the impact of attack and detection and prevention mechanism in terms of throughput. As shown in the line graph, under the presence of black hole Attack the throughput was almost equal to zero. But with detection, the throughput of proposed DTH-AODV was very much improved and almost equal to that of AODV without any attack. In terms of throughput, the proposed DTH-AODV performed a little bit better than previous PTH-AODV.

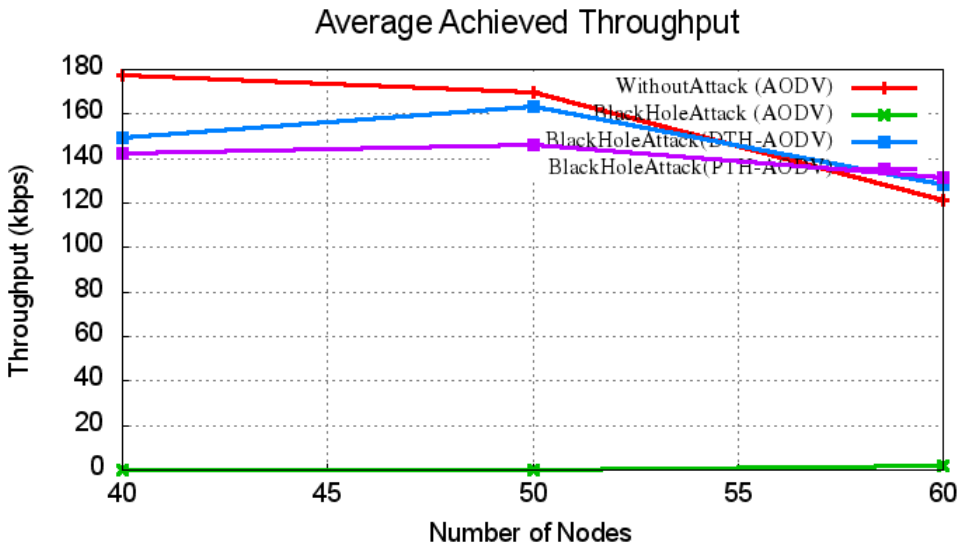

Figure 5.9: Comparison of Network Size Vs Throughput in DTH-AODV.

As shown in the line graph figure 5.10, under the presence of black hole attack the PDF was almost equal to zero. And at low network density PDF is equal to zero. For example, at 40 nodes, it is zero because, among the 40 nodes, 15 are malicious- so that they will able to break all the communication between other nodes. But with detection, the PDF of proposed DTH-AODV was very much improved and almost equal to that of AODV without any attack. In terms of PDF, the performance of AODV, \proposed DTH-AODV and previous PTH-AODV are almost equal.

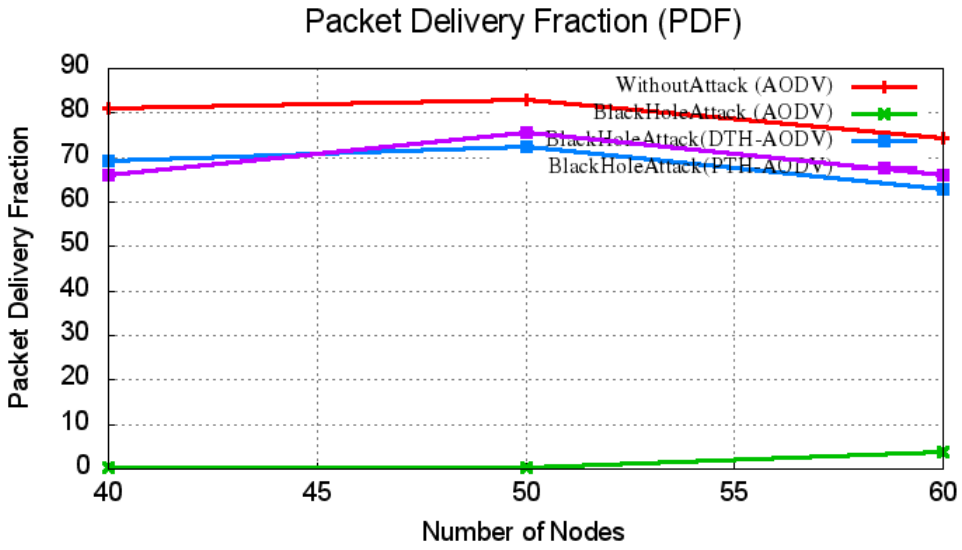

Figure 5.10: Comparison of Network Size Vs PDF in DTH-AODV.

The graph in figure 5.11 shows the impact of attack and detection and prevention mechanism in terms of End-to-end Delay (EED) of data flows. With respect to the increase of no of nodes in the network, the performance getting decreased .As shown in the line graph, black hole attack seems to be providing lower EED than AODV(without attack) – but certainly it does not mean that black hole attack is improving the performance of the network. The low EED under attack is due to a strange fact that the attack makes disconnection in TCP flows and since the packets are not at all forwarded to any further nodes, indirectly it is reduce the message overhead in the network and reduced bandwidth usage otherwise it will be consumed by the forwarded data packets. So, the flows that were unaffected by black hole attack (the connections where there is no neighboring attack nodes) utilizes that extra bandwidth and gains some performance in term of some metrics. In terms of EED, the proposed DTH-AODV performed a little bit better than previous PTH-AODV.

The EED of DTH-AODV was a little bit higher than AODV. Because, under attack detection and prevention, alternate route will be resolved by avoiding malicious nodes on a path, So that the path length will get increased and hence will increase the EED.

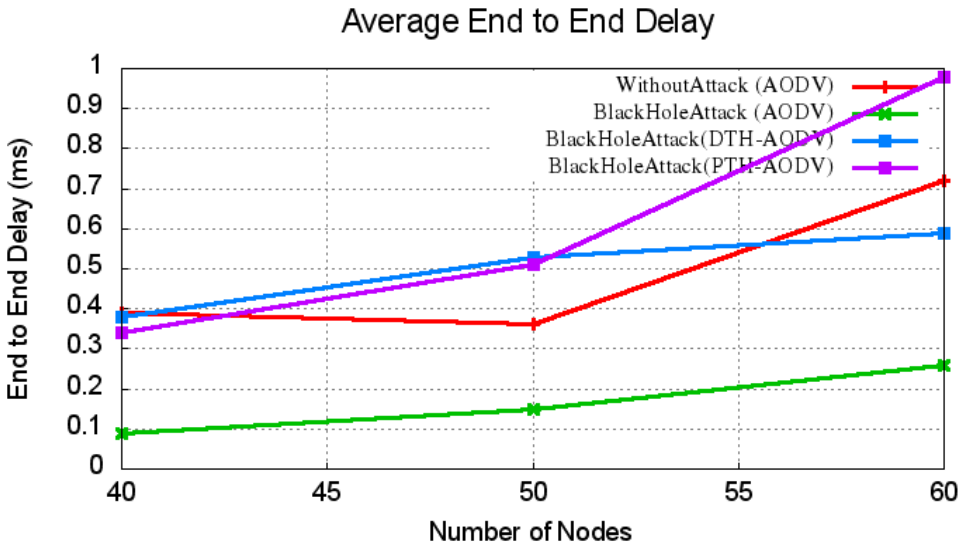

Figure 5.11: Comparison of Network Size Vs End-to-end Delay in DTH-AODV.

The graph in figure 5.12 shows the impact of attack and detection and prevention mechanism in terms of consumed battery energy. As shown in the line graph, in the

presence of Attack the battery consumption is lesser than AODV (without attack) – but certainly it does not mean these Attacks are improving the performance in terms of energy consumption. The low energy consumption under attacks are due to a strange fact that these attacks makes disconnection in data flows and since the packets are not at all forwarded to any further nodes, indirectly it is reduce the battery consumption at the other nodes otherwise it will be consumed for forwarding the data packets. So, the nodes that were unaffected by Attacks (where there is no neighboring attack nodes) preserves some battery power. Understanding this strange fact requires a better visualization of the whole network scenario. It is simple – without any attack, AODV was able to send much and maximum nodes were able to participate in that communication and utilized their energy for transmission/forwarding of packets – so that the energy is consumed in most of the nodes. But in the presence of attack, the packets are getting dropped intermediately and the battery powers on other nodes that are not at all forwarding the packets gets preserved. With respect to the increase of no of nodes in the network, the performance seems to be getting decreasing.

But, interestingly, the energy consumption in the case of proposed DTH-AODV is a little bit lesser than AODV. This obviously proves the better working of proposed detection model. In terms of consumed energy, the proposed DTH-AODV consumed a little bit high energy than previous PTH-AODV – this is because, the DTH-AODV will try to send more packets than PTH-AODV.

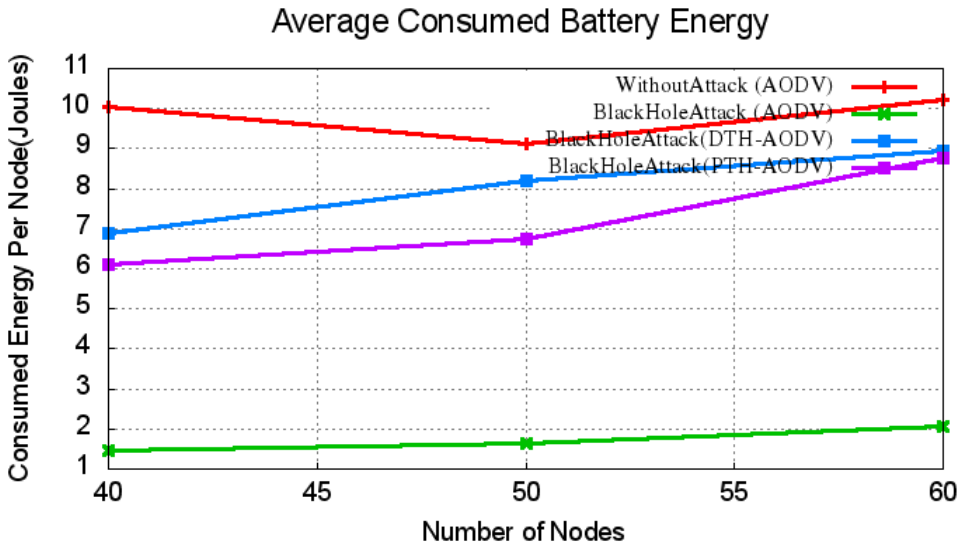

Figure 5.12: Comparison of Network Size Vs Battery Energy in DTH-AODV.

The graph in figure 5.13 shows the impact of attack and detection and prevention mechanism in terms of overhead. As shown in the line graph, under the presence of Black hole the overhead is minimum – because, the black holes just breaks all the communication. But with proposed DTH-AODV based detection and prevention mechanism, the overhead becomes equal to that of AODV – it signifies that the proposed DTH-AODV works almost equal to AODV. In terms of overhead, the proposed DTH-AODV imposed a little bit high overhead than previous PTH-AODV – this is because; the DTH-AODV will try to send more packets than PTH-AODV.

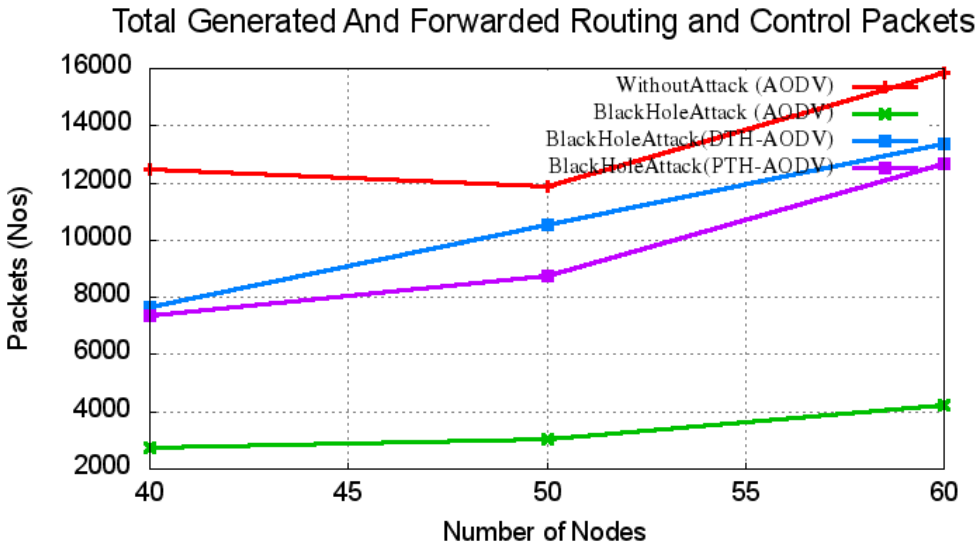

Figure 5.13: Comparison of Network Size Vs Overhead in DTH-AODV.

The graph in figure 5.14 shows the impact of attack and detection and prevention mechanism in terms of total maliciously dropped packets at MAC layer. For the first look, one may think as this as a wrong result because of the decrease in malicious dropping in the case of attack as well as detection and prevention (PTH-AODV). But it is not. The dropping in the case of black hole attack is decreased because, the malicious packet dropping is only happening at routing layer. The dropping in the case of attack is less than all because, PTH-AODV a little bit higher than attack without detection because, PTH-AODV will try to avoid black holes so that, initiate new route discovery process and this causes more packet generation and loss at MAC layer. In terms of MAC Layer dropped packets, the performance of proposed DTH-AODV a little poor than previous PTH-AODV – this is because, the DTH-AODV will try to send more packets than PTH-AODV and hence the packet loss at MAC layer also increases a little bit.

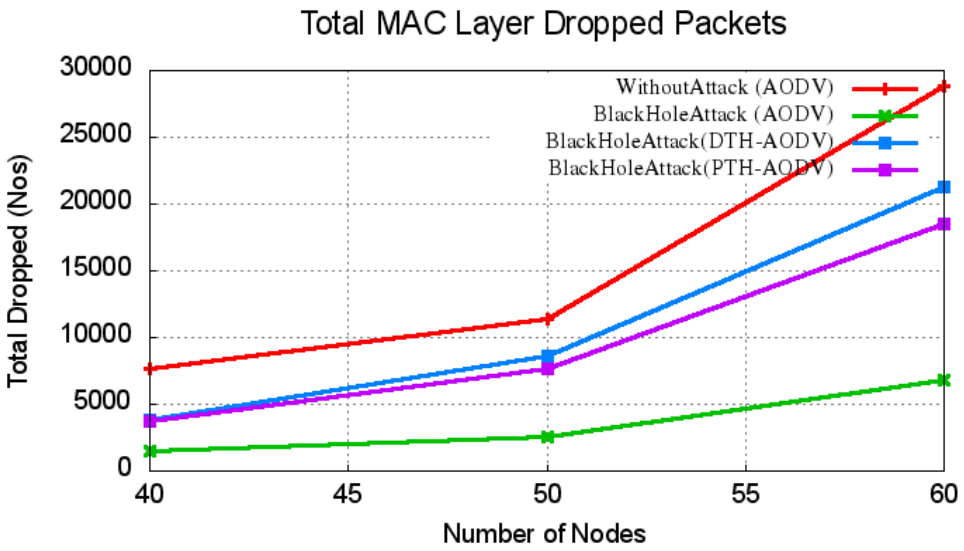

Figure 5.14: Comparison of Network Size Vs MAC Layer Dropped in DTH-AODV.

The graph in figure 5.15 shows the impact of attack and detection and prevention mechanism in terms of total maliciously dropped packets at network layer. For the first look, one may think as this as a wrong result because of the increase in malicious dropping in the case of detection and prevention (DTH-AODV). But it is not. The malicious dropping in the case of DTH-AODV is increased because it is trying to send the packet in one way or another by avoiding malicious nodes.

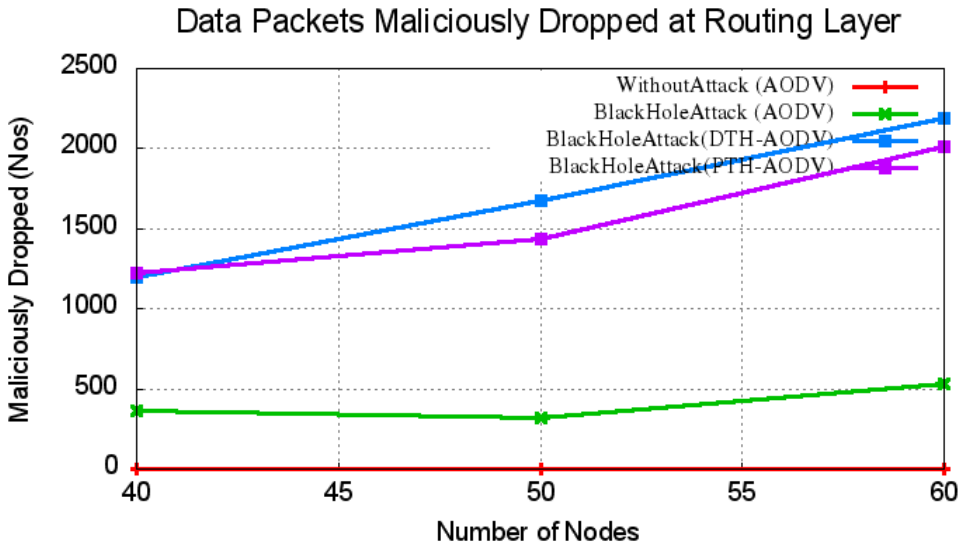

Figure 5.15: Comparison of Network Size Vs Malicious Drops at Routing Layer in DTH-AODV.

The retransmissions involved in this process increases malicious packet dropping. In terms of Routing Layer maliciously dropped packets, the performance of proposed DTH-AODV a little poor than previous PTH-AODV – this is because, the DTH-AODV will try to send more packets than PTH-AODV and hence the packet loss at MAC layer also increases a little bit.

5.3 Comparison of performance of Periodic and Dynamic Trust Handshake Based Malicious Behavior Detection Mechanisms (P / DTH-AODV) with Selfish node Attack and Black Hole Attack

After having discussed PTH-AODV and DTH-AODV, the two protocols are compared against selfish node attack and black hole attack. Following tables first presents the trace files and then graphs depict the comparison.

Table 5.1: Trace file of modified AODV with variable nodes

Nodes	PDF	NRL	EED	Overhd	SDDropped	Throughput	MAC Load	ConsEnergy	MalDropped	MAC Dropped	Sent	Received
40	81.00	8.79	0.39	12470	333.00	177.45	27.91	10.04	0.00	7609.0	1751.0	1418.0
50	83.00	8.43	0.36	11903	290.00	170.07	24.78	9.13	0.00	11416.	1702.0	1412.0
60	74.40	14.37	0.72	15862	380.00	121.47	39.87	10.22	0.00	28864.	1484.0	1104.0

Table 5.2: Trace file of AODV with Black hole Attack

Nodes	PDF	NRL	EED	Overhd	SDDropped	Throughput	MAC Load	ConsEnergy	MalDropped	MAC Dropped	Sent	Received
40	0.30	909.33	0.09	2728.0	1017.0	0.10	2425.0	1.46	364.00	1511.0	1020.0	3.00
50	0.20	1511.5	0.15	3023.0	1018.0	-0.00	3965.5	1.65	317.00	2565.0	1020.0	2.00
60	3.70	110.79	0.26	4210.0	982.00	1.74	293.18	2.06	536.00	6786.0	1020.0	38.00

Table 5.3: Trace file of AODV with Selfish node Attack

Nodes	PDF	NRL	EED	Overhd	SDDropped	Throughput	MAC Load	ConsEnergy	MalDropped	MAC Dropped	Sent	Received
40	53.40	8.30	0.13	6456.0	680.00	102.56	21.93	4.17	394.00	4144.0	1458.0	778.00
50	60.40	9.47	0.15	8382.0	581.00	110.70	24.85	5.30	562.00	9138.0	1466.0	885.00
60	57.60	11.71	0.17	10457.	657.00	120.03	30.73	7.07	445.00	17991.	1550.0	893.00

Table 5.4: Trace file of DTH-AODV with Black hole Attack

Nodes	PDF	NRL	EED	Overhd	SDDropped	Throughput	MAC Load	ConsEnergy	MalDropped	MAC Dropped	Sent	Received
40	69.40	6.67	0.38	7654.0	507.00	149.68	22.37	6.88	1201.0	3857.0	1655.0	1148.0
50	72.30	8.44	0.53	10561.	479.00	163.63	25.76	8.20	1674.0	8565.0	1730.0	1251.0
60	62.90	13.42	0.59	13392.	588.00	128.02	37.22	8.95	2186.0	21234.	1586.0	998.00

Table 5.5: Trace file of DTH-AODV with Selfish node Attack

Nodes	PDF	NRL	EED	Overhd	SDDropped	Throughput	MAC Load	ConsEnergy	MalDropped	MAC Dropped	Sent	Received
40	64.20	7.87	0.48	7578.0	537.00	118.74	22.54	5.60	224.00	4589.0	1500.0	963.00
50	71.40	7.22	0.29	8687.0	481.00	157.30	19.86	6.60	61.00	8560.0	1684.0	1203.0
60	57.40	12.53	0.47	10436.	618.00	101.81	30.35	6.57	131.00	20823.	1451.0	833.00

Table 5.6: Trace file of PTH-AODV with black hole attack

Nodes	PDF	NRL	EED	Overhd	SDDropped	Throughput	MAC Load	ConsEnergy	MalDropped	MAC Dropped	Sent	Received
40	66.20	6.82	0.34	7362.0	550.00	142.40	22.21	6.10	1219.0	3726.0	1629.0	1079.0
50	75.60	7.25	0.51	8756.0	389.00	146.23	21.52	6.74	1432.0	7624.0	1596.0	1207.0

60	66.00	12.05	0.98	12669.	541.00	131.60	35.23	8.78	2016.0	18490.	1592.0	1051.0

Table 5.7: Trace file of PTH-AODV with selfish node attack

Nodes	**PDF**	**NRL**	**EED**	**Overhd**	**SDDropped**	**Throughput**	**MAC Load**	**ConsEnergy**	**MalDropped**	**MAC Dropped**	**Sent**	**Received**
40	63.70	6.97	0.43	6436.0	525.00	111.35	20.27	4.49	140.00	3808.0	1448.0	923.00
50	66.20	8.45	0.28	8721.0	527.00	132.80	22.65	6.10	121.00	7854.0	1559.0	1032.0
60	52.90	14.52	0.31	11138.	682.00	99.71	33.69	6.19	76.00	22121.	1449.0	767.00

5.3.1 Analysis of Results With Respect to Different Network Size

Here we see the analytic results of comparison of black hole attacks with AODV (it means performance without any attack). And it is studied with Respect to Different Network Size. In the following analysis the total number of nodes in the network is varied as 40, 50 and 60 and among them, the number of malicious nodes kept as 15 and the impact is measured using different metrics.

The graph in figure 5.16 shows the impact of attack and detection and prevention mechanism in terms of total data packets sent at application source. As shown in the line graph, under the presence of black hole attack, the application source itself can not able to send much. But while detection the proposed DTH-AODV was able to send as much as AODV without any attack. In terms of send packets, the proposed DTH-AODV performed a little bit better than previous PTH-AODV. Further, under the selfish node attack, the algorithms PTH-AODV and DTH-AODV provided considerable improvement in terms of sent packets.

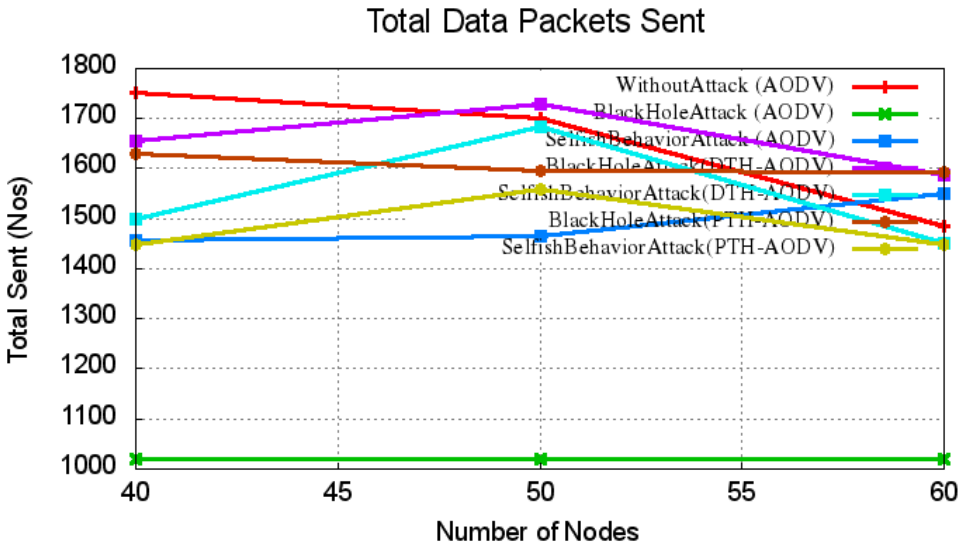

Figure 5.16: Comparison of Network Size Vs Sent Packets in PTH-AODV and DTH-AODV.

The graph in figure 5.17 shows the impact of attack and detection and prevention mechanism in terms of total data packets received at application destination. As shown in the line graph, under the presence of black hole attack the application destination itself can not able to receive anything. But while detection the proposed DTH-AODV was able to receive as much as AODV without any attack. In terms of received packets, the proposed DTH-AODV performed a little bit better than previous PTH-AODV. Further, under the selfish node attack, the algorithms PTH-AODV and DTH-AODV provided considerable improvement in terms of received packets.

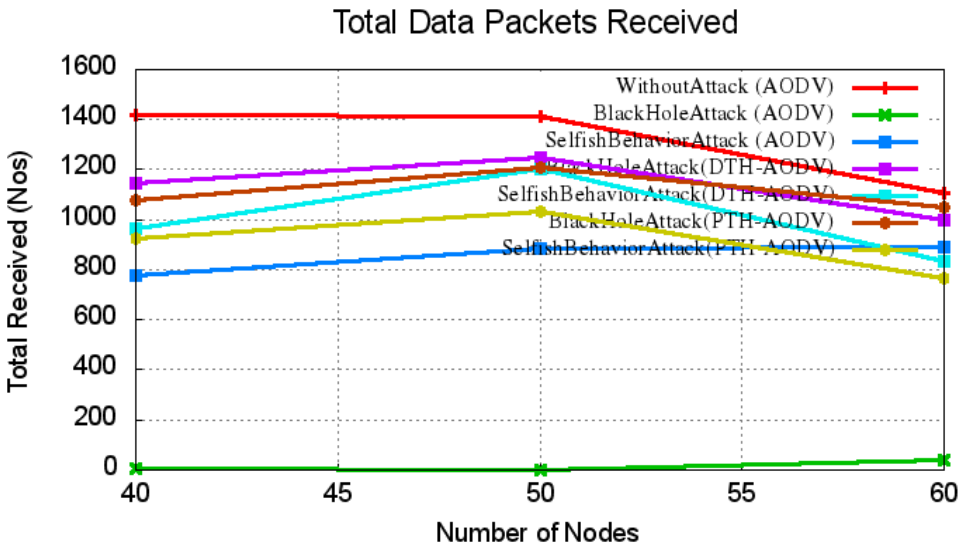

Figure 5.17: Comparison of Network Size Vs Received Packets in PTH-AODV and DTH-AODV.

The graph in figure 5.18 shows the impact of attack and detection and prevention mechanism in terms of routing load. As shown in the line graph, under the presence of Black hole the routing load is very high. But with proposed DTH-AODV based detection

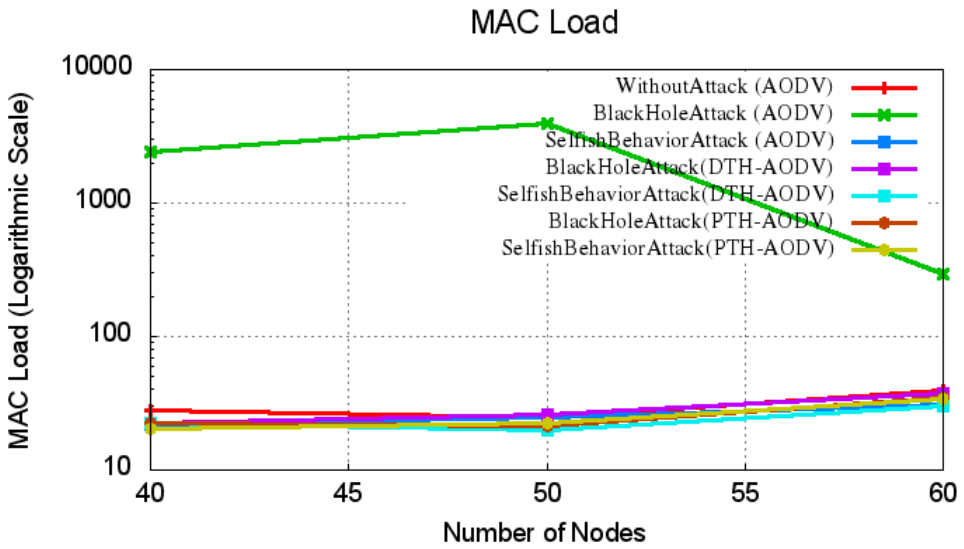

Figure 5.18: Comparison of Network Size Vs Routing Load in PTH-AODV and DTH-AODV.

and prevention mechanism, the routing load was almost equal to that of AODV. In terms of routing load, the performance of AODV, proposed DTH-AODV and previous PTH-AODV are almost equal under the selfish node attack as well as black hole attack.

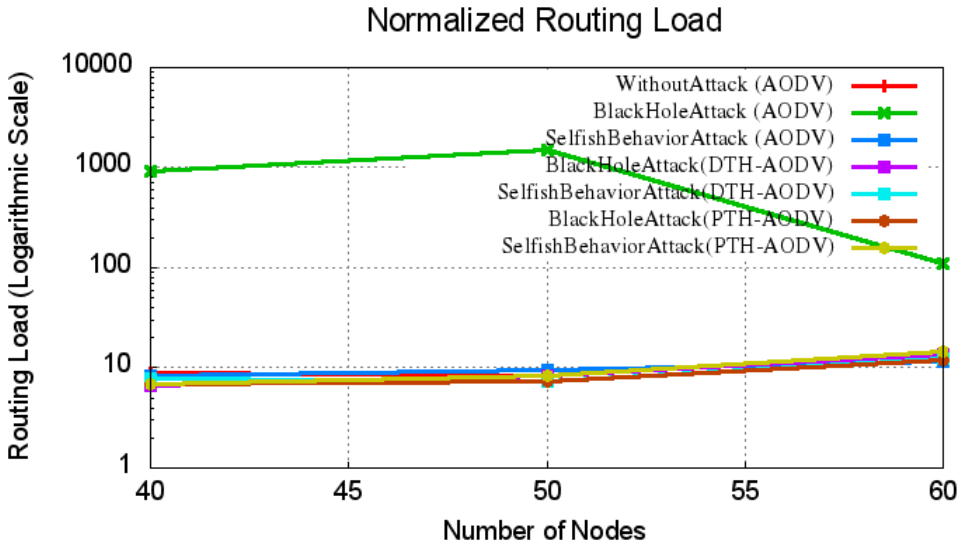

Figure 5.19: Comparison of Network Size Vs MAC Load in PTH-AODV and DTH-AODV.

The graph in figure 5.19 shows the impact of attack and detection and prevention mechanism in terms of MAC load. As shown in the line graph, under the presence of

Black hole the MAC load is very high. But with proposed DTH-AODV based detection and prevention mechanism, the MAC load was almost equal to that of AODV. In terms of MAC load, the performance of AODV, proposed DTH-AODV and previous PTH-AODV are almost equal under the selfish node attack as well as black hole attack.

The graph in figure 5.20 shows the impact of attack and detection and prevention mechanism in terms of total dropped packets at the application layer. As shown in the line graph, under the presence of black hole attack a lot of packets were dropped at the application layer. But while detection, the packet dropping of proposed DTH-AODV was very much reduced and al most equal to that of AODV without any attack. In terms of the application layer dropped packets, the proposed DTH-AODV dropped a little bit high number of packets than previous PTH-AODV – this is because, the DTH-AODV will try to send more packets than AODV and PTH-AODV.

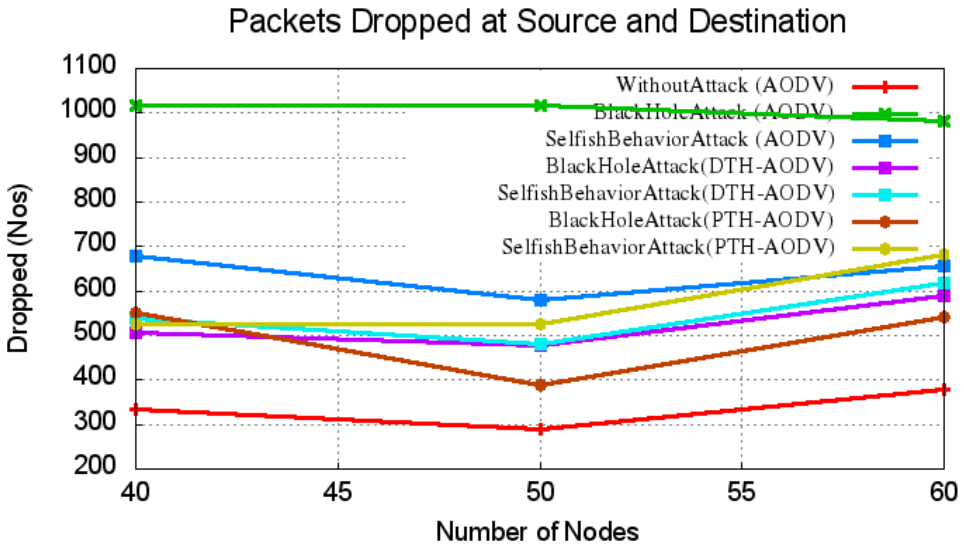

Figure 5.20: Comparison of Network Size Vs Packets Dropped At The application layer in PTH-AODV and DTH-AODV.

The graph in figure 5.21 shows the impact of attack and detection and prevention mechanism in terms of throughput. As shown in the line graph, under the presence of black hole attack the throughput was almost equal to zero. But with detection, the throughput of proposed DTH-AODV was very much improved and almost equal to that of AODV without any attack. In terms of throughput, the proposed DTH-AODV performed a little bit better than previous PTH-AODV. Further, under the selfish node

attack, the algorithms PTH-AODV and DTH-AODV provided considerable improvement in terms of throughput.

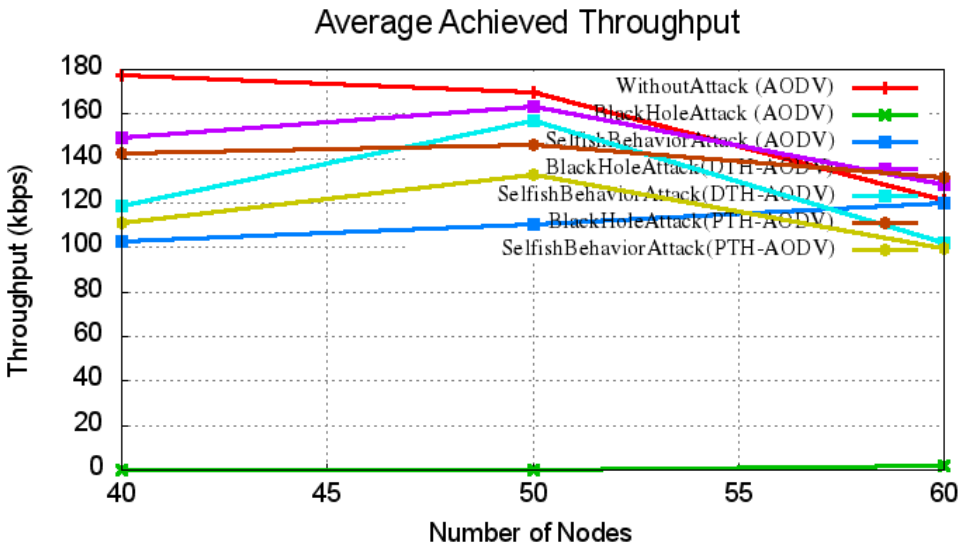

Figure 5.21: Comparison of Network Size Vs Throughput in PTH-AODV and DTH-AODV.

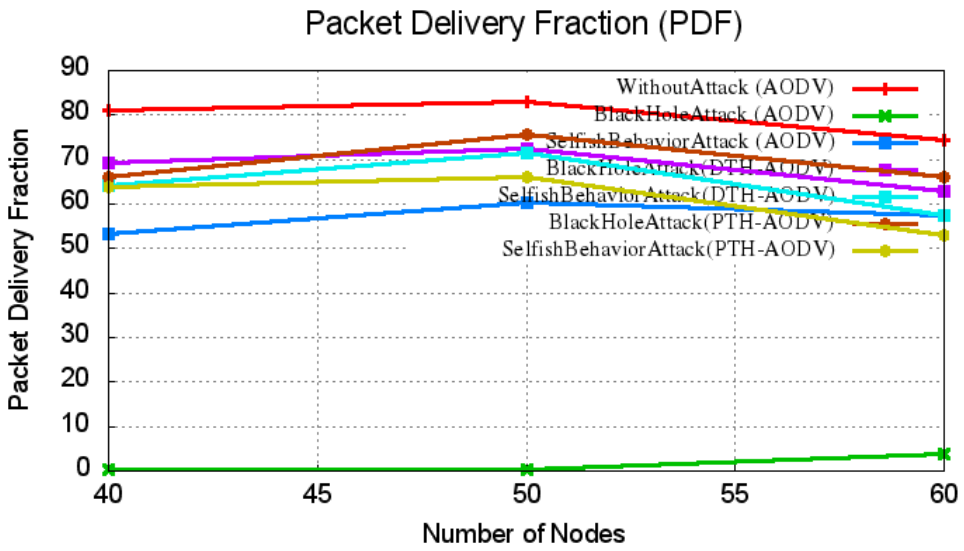

Figure 5.22: Comparison of Network Size Vs PDF in PTH-AODV and DTH-AODV.

The graph in figure 5.22 shows the impact of attack and detection and prevention mechanism in terms of PDF. As shown in the line graph, under the presence of black hole attack the PDF was almost equal to zero. And at low network density PDF is equal to zero. For example, at 40 nodes, it is zero because, among the 40 nodes, 15 are malicious- so that they will able to break all the communication between other nodes. But with

detection, the PDF of proposed DTH-AODV was very much improved and almost equal to that of AODV without any attack. In terms of PDF, the performance of AODV, proposed DTH-AODV and previous PTH-AODV are almost equal. Further, under the selfish node attack, the algorithms PTH-AODV and DTH-AODV provided considerable improvement in terms of PDF.

The graph in figure 5.23 shows the impact of attack and detection and prevention mechanism in terms of End-to-end Delay (EED) of data flows. With respect to the increase of no of nodes in the network, the performance getting decreased .As shown in the line graph, black hole attack seems to be providing lower EED than AODV(without attack) – but certainly it does not mean that black hole attack is improving the performance of the network. The low EED under attack is due to a strange fact that the attack makes disconnection in TCP flows and since the packets are not at all forwarded to any further nodes, indirectly it is reduce the message overhead in the network and reduced bandwidth usage otherwise it will be consumed by the forwarded data packets. So, the flows that were unaffected by black hole attack (the connections where there is no neighboring attack nodes) utilizes that extra bandwidth and gains some performance in term of some metrics. In terms of EED, the proposed DTH-AODV performed a little bit better than previous PTH-AODV.

The EED of DTH-AODV was a little bit higher than AODV. Because, under attack detection and prevention, alternate route will be resolved by avoiding malicious nodes on a path, So that the path length will get increased and hence will increase the EED.

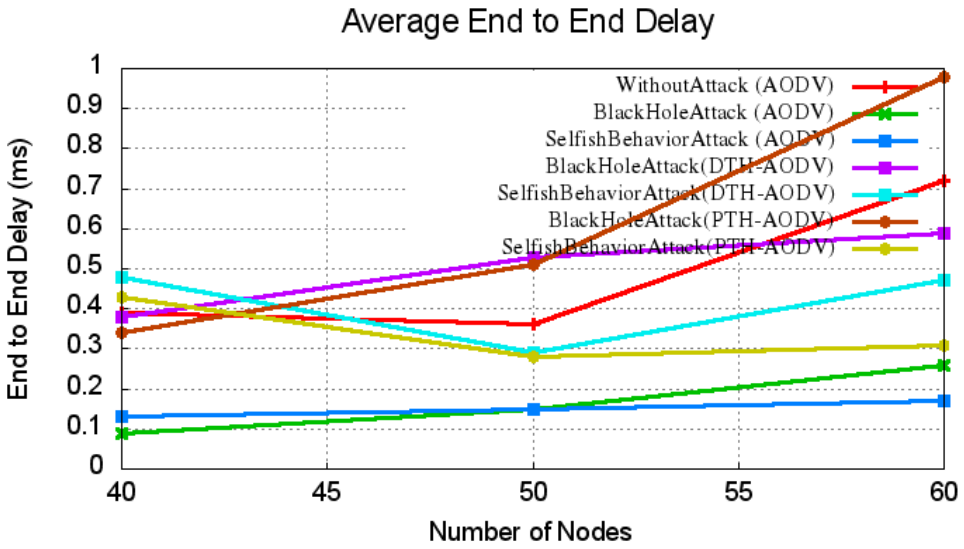

Figure 5.23: Comparison of Network Size Vs End-to-end Delay in PTH-AODV and DTH-AODV.

The graph in figure 5.24 shows the impact of attack and detection and prevention mechanism in terms of consumed battery energy. As shown in the line graph, in the presence of Attack the battery consumption is lesser than AODV (without attack) – but certainly it does not mean these Attacks are improving the performance in terms of energy consumption. The low energy consumption under attacks are due to a strange fact that these attacks makes disconnection in data flows and since the packets are not at all forwarded to any further nodes, indirectly it is reduce the battery consumption at the other nodes otherwise it will be consumed for forwarding the data packets. So, the nodes that were unaffected by Attacks (where there is no neighboring attack nodes) preserves some battery power. Understanding this strange fact requires a better visualization of the whole network scenario. It is simple – without any attack, AODV was able to send much and maximum nodes were able to participate in that communication and utilized their energy for transmission/forwarding of packets – so that the energy is consumed in most of the nodes. But in the presence of attack, the packets are getting dropped intermediately and the battery powers on other nodes that are not at all forwarding the packets gets preserved. With respect to the increase of no of nodes in the network, the performance seems to be getting decreasing.

But, interestingly, the energy consumption in the case of proposed DTH-AODV is a little bit lesser than AODV. This obviously proves the better working of proposed detection model. In terms of consumed energy, the proposed DTH-AODV consumed a little bit high energy than previous PTH-AODV – this is because, the DTH-AODV will try to send more packets than PTH-AODV.

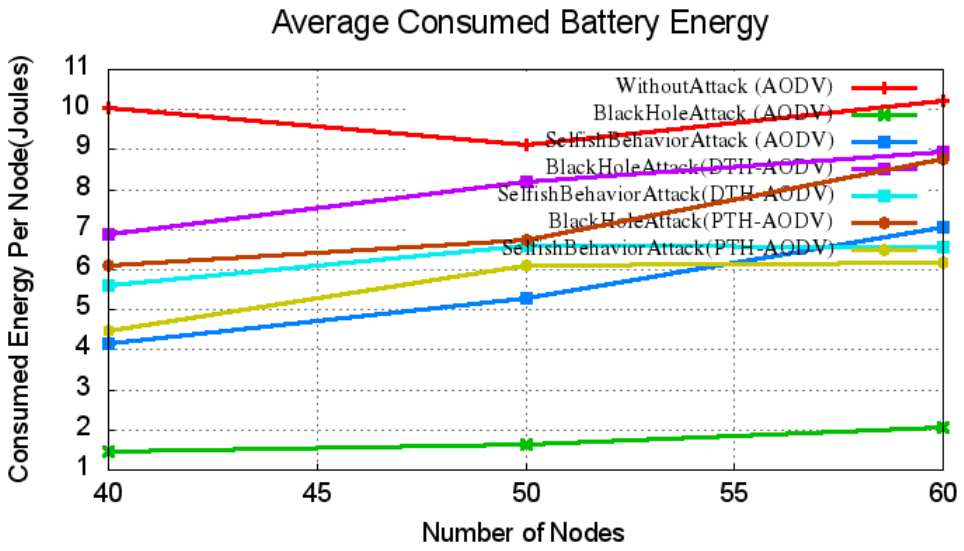

Figure 5.24: Comparison of Network Size Vs Battery Energy in PTH-AODV and DTH-AODV.

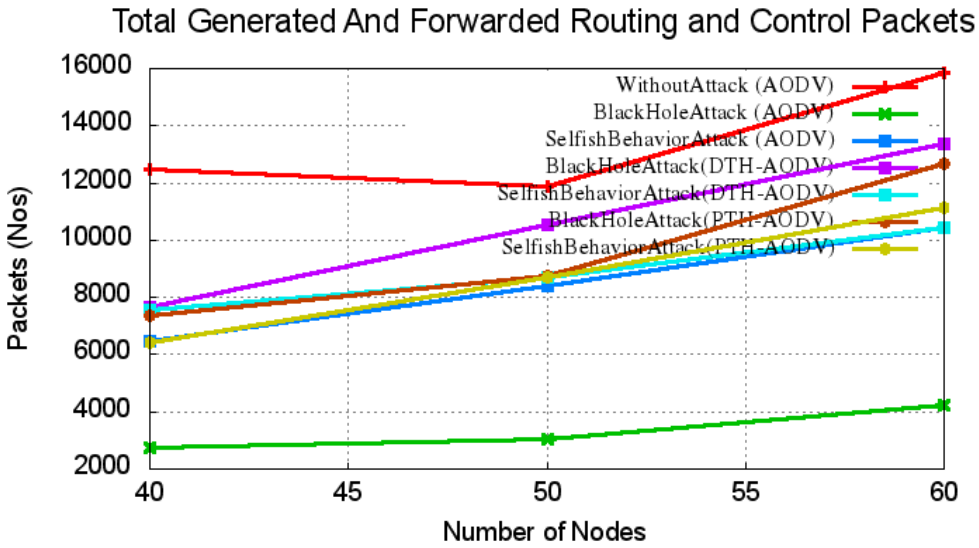

Figure 5.25: Comparison of Network Size Vs Overhead in PTH-AODV and DTH-AODV.

The graph in figure 5.25 shows the impact of attack and detection and prevention mechanism in terms of overhead. As shown in the line graph, under the presence of black hole the overhead is minimum – because, the black holes break all the communication.

But with proposed DTH-AODV based detection and prevention mechanism, the overhead becomes equal to that of AODV – it signifies that the proposed DTH-AODV works almost equal to AODV. In terms of overhead, the proposed DTH-AODV imposed a little bit high overhead than previous PTH-AODV – this is because the DTH-AODV will try to send more packets than PTH-AODV.

The graph in figure 5.26 shows the impact of attack and detection and prevention mechanism in terms of total maliciously dropped packets at MAC layer. The dropping in the case of black hole attack is decreased because, the malicious packet dropping is only happening at routing layer. The dropping in the case of attack is less than all because, PTH-AODV a little bit higher than attack without detection because, PTH-AODV will try to avoid black holes so that, initiate new route discovery process and this causes more packet generation and loss at MAC layer. In terms of MAC Layer dropped packets, the performance of proposed DTH-AODV a little poor than previous PTH-AODV – this is because, the DTH-AODV will try to send more packets than PTH-AODV and hence the packet loss at MAC layer also increases a little bit.

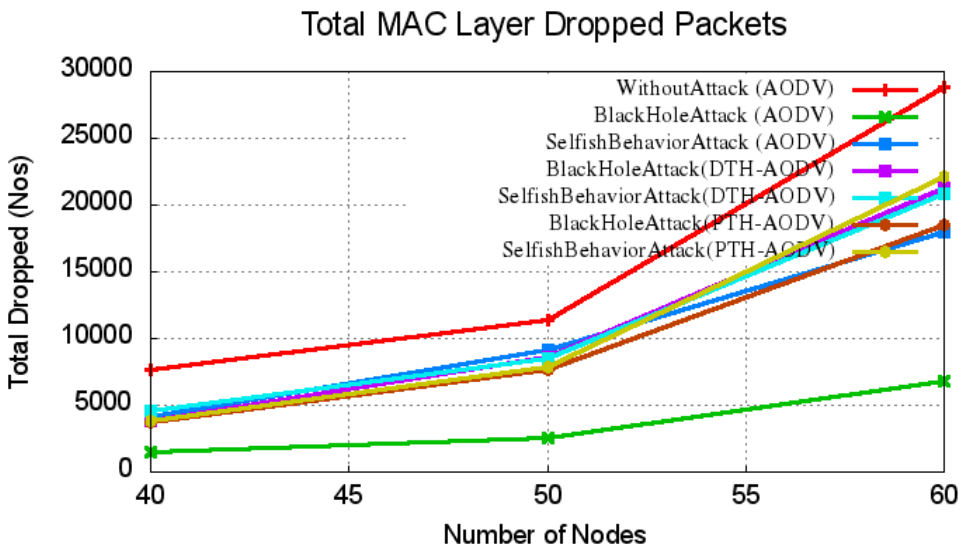

Figure 5.26: Comparison of Network Size Vs MAC Layer Dropped in PTH-AODV and DTH-AODV.

The graph in figure 5.27 shows the impact of attack and detection and prevention mechanism in terms of total maliciously dropped packets at network layer. For the first look, one may think as this as a wrong result because of the increase in malicious dropping in the case of detection and prevention (DTH-AODV). But it is not. The

malicious dropping in the case of DTH-AODV is increase because, it is trying to send the packet in one way or another by avoiding malicious nodes. The retransmissions involved in this process increases malicious packet dropping. In terms of Routing Layer maliciously dropped packets, the performance of proposed DTH-AODV a little poor than previous PTH-AODV this is because, the DTH-AODV will try to send more packets than PTH-AODV and hence the packet loss at MAC layer also increases a little bit.

Even though the PTH-AODV and DTH-AODV detection and prevention methods improved the performance under selfish node attack, that improvement is only minimum if we compare it with the improvement under black hole attack.

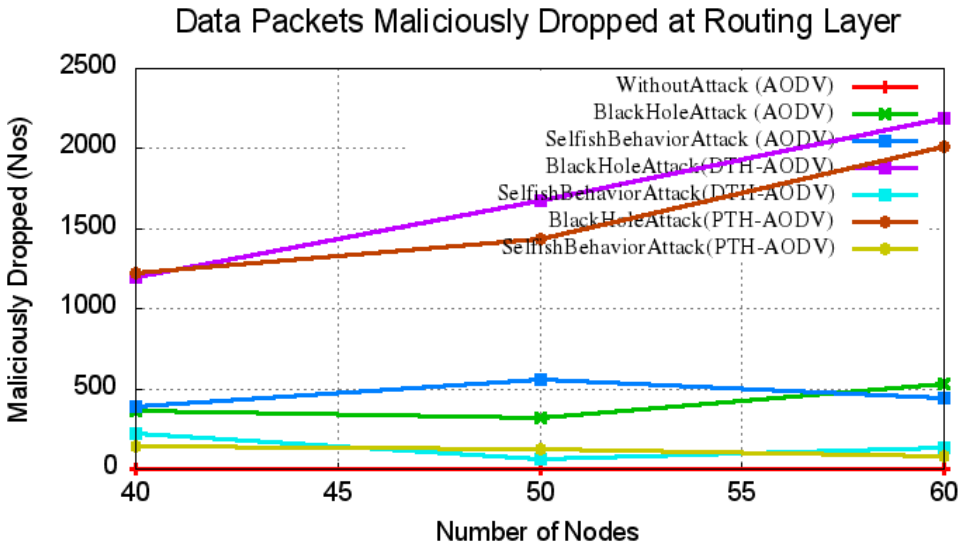

Figure 5.27: Comparison of Network Size Vs Malicious Drops at Routing Layer in PTH-AODV and DTH-AODV.

The reason is, under the presence of 15 black hole nodes, technically, the black hole attack on those 15 nodes will entirely affect all the communications in the network. Because, each black hole will just collapse all its neighbor's routing table with false information so that will collapse all the proceeding TCP and UDP connections. But with 15 selfish nodes in the network, the communication was not that much affected. Because, in selfish nodes only affect the communications that were happening through it and will not affect other nodes.

5.4 Comparison of PTH-AODV and DTH-AODV with other Trust Based Routing Algorithm

The table 5.8 compares the characteristics of conventional AODV based trust routing algorithm with newly developed algorithm PTH-AODV and its modified version DTH-AODV.

Table 5.8: Comparison of characteristics of existing AODV based trust routing protocol with PTH-AODV and DTH-AODV

Secure Routing Algorithm	Characteristics	Advantages	Disadvantages
Collaborative Trust-Based Secure Routing against Colluding Malicious Nodes (T. Ghosh et al. [165])	This Protocol design assumes the prior distribution of trust to all the nodes. This protocol also assumes the presence of public key infrastructure because whenever the node transmits RREQ message containing trust metric to intermediate node, the next node authenticates the previous node by signing with its private key.	This protocol is highly resistant towards attack where a malicious node claims to have genuine identity such as	This protocol aims to find the shortest path to the destination node irrespective of presence of malicious node therefore it is more susceptible to internal attacks. The prior distribution of trust makes the network less dynamic and adaptable to changing situations. The Use of Public key infrastructure makes the protocol highly expensive to use and it also causes more

			overhead to maintain all the keys. This protocol fails under the situation of compromised node.
Trust-Embedded AODV (T-AODV) (T. Ghosh et al. [166])	This Protocol is an extension of 131 with the difference that the trust factor is periodically updated by the exchange of routing messages. However this protocol also assumes the existence of public key infrastructure and it also assumes the radio rage of all the node are same which is more theoretical to study.	In this protocol the following actions done by the malicious node is avoided. If the malicious node provides the wrong information in the RREQ packet by changing the hop count field r the destination address etc. If the malicious node decrypts the sign given by the genuine node with the intention to alter the information given in the header. It is more adaptale to topology changes	This protocol faikls to find the secure end-to-end path from source to destination. More overhead of public key infrastructure.
Trust Establishment in Pure Ad-hoc Networks [1168]	This protocol does not require trusted third party	Malicious node are bypassed during route discoveries.	Extra overhead in added due to nature of the protocol.

	infrastructure for its operation. All node computer the trust value based on direct feedback	This protocol achieves better throughput in presence of malicious node.	The accuracy of protocol depends on the weight values that are assigned in the calculation of trust values. This protocol is more susceptible to IP spoofing attack and MAC spoofing attack. This protocol fails when the malicious node collude.
Opinion Based Trusted Routing Protocol – TAODV [169]	This Protocol uses soft encryption technique.	The encrypted parts of message are forwarded through different routes so malicious node hardly have access to complete message	This protocol is suceptibe to internal attack. It takes more time in route selection. It is also possible not to route all the messages securely
Friendship based routing algorithm – frAODV [159]	Each node stores the list of friends nodes and friendship value. The friendship value determines the level of trustworthiness. During control packet transmission the friendship value	The performance gives better results for the more dynamic network.	The experiment is performed based on 5 number of nodes so the performance of protocol for the large number of node is undetermined.

	is also exchanged between the nodes.		
PTH-AODV	This protocol does not require public key infrastructure. It also do not use any encryption technique. The Trust value is periodically exchanged by the nodes and the trust value depends on the feedback given by the previous neighbour in the successful communication.	This protocol detects all the selfish and malicious node due to exchange of trust value between the nodes. Moreover sometimes some genuine node that are not participating in the communication for over a long time are falsely interpreted as the selfish node. This protocol also avoids false accusation of genuine node as the selfish node.	This protocol causes more packet overhead as the trust factor is periodically exchanged by the nodes in the network.
DTH-AODV	This protocol does not require public key infrastructure. It also do not use any encryption technique. The However the Trust value is exchanged by the nodes only	Additional overhead caused by periodic exchange of trust metric is also avoided. This protocol detects all the selfish and malicious node due	More work can be done in case of trust dispersal and trust decay over time. Trust can also be gathered by the malicious scenarios. More work can also be done in case of

	when there is a need of route establishment and the trust value depends on the feedback given by the previous neighbour in the successful communication.	to exchange of trust value between the nodes. Moreover sometimes some genuine node that are not participating in the communication for over a long time are falsely interpreted as the selfish node. This protocol also avoids false accusation of genuine node as the selfish node.	malicious colluding nodes.

5.5 Summary

In this chapter, dynamic trust handshake based detection of black hole attack is proposed. DTH-AODV is implemented under NS2 and its performance is compared with the results of previous PTH-AODV, AODV and AODV under attack. As shown in the results of the previous section, the proposed DTH-AODV improved the throughput and PDF almost equal to that of AODV and performed a little bit better than previously proposed PTH-AODV.

Further, a two trust handshake based detection methods for detecting black hole attack and selfish node attack is proposed. DTH-AODV under NS2 is implemented and compared its performance with the results of previous PTH-AODV, AODV and AODV under attack. The main advantage of the proposed DTH-AODV is : it will detect and prevent the malicious nodes in the very early stage of route discovery process. So, it will not need any manipulation in routing tables in the route resolving process, because, by the

design, it will avoid including malicious hops in routing table of normal nodes at the route discovery process itself.

According to the arrived results, dynamic trust handshake based malicious node detection and prevention mechanism worked good and successfully detected black hole nodes in the network and avoided establishing routes though them. As shown in the results of the previous section, the proposed DTH-AODV improved the throughput and PDF almost equal to that of AODV and performed a little bit better than previously proposed PTH-AODV.

As shown in the graphs of previous section, even though the proposed detection and prevention methods improved the performance in the case of selfish node attack, that improvement is not significant as in the case of black hole attack and even worse in case of jellyfish attack.

Chapter-6

Conclusion and Future Scope

The topic of my present study is of immense importance with the basic objective of research work on the design, development of an efficient secure AODV routing protocol for MANETs. Security is always a concern during the routing of the packets from one node to another (mutihop), when the packets travel from source node to the sink node. An exhaustive literature survey has been carried out before developing a secure and efficient routing algorithm. The various attacks that are shortlisted based upon their three different classifications - Selfish node attack, is a passive attack (classified under evesdropping); Black hole attack is an active attack (classified under dropped data packet) and jellyfish attack is also an active attack (classified under modification attack). The comprehensive experimental analysis based on simulations using NS2, has been carried out to evaluate the performance of two proposed secure and efficient routing algorithms, namely, Periodic AODV (PTH-AODV) and Dynamic AODV (DTH-AODV) This chapter summarizes the significant conclusions obtained and suggests some future direction for further work.

7.1 Conclusion

Among the various state-of-the-art security approaches, trust based approaches are promising due to their lesser overhead in key maintenance and the ability to keep pace with the dynamic nature of the MANETs. Some of the significant outcomes are presented below:

1. Analysis of Attacks on AODV Routing Protocol: A comparative analysis of three kinds of Jellyfish Attacks and Black Hole Attack with selfish node attack under AODV

routing protocol is presented. The analysis is made with respect to i) different network sizes ii) under the presence of the different number of attackers in the network.

a) The performance of AODV protocol is analyzed first of all by varying the size of the network as 40, 50 and 60 in the NS2.35. It is seen that black hole attack affects the total sent packets and total received packet as the application source itself are not able to send much. selfish node attack seems to be causing a little bit higher impact than all the Jellyfish Attacks. With respect to the increase of the number of attackers, the performance decreases. Moreover, in case of packet delivery ratio, black hole attack caused much packet loss so that the PDF was very lower than all other attacks. Next to black hole attack, Selfish node attack seems to be causing a little bit higher impact than all the Jellyfish Attacks. In the presence of black hole attack and selfish node attack the battery consumption is lesser than all other Jellyfish Attacks.

b) The next analysis of attacks is done by varying the number of malicious nodes as 10, 15 and 20 by keeping the number of nodes constant as 40. With respect to the increase of number of nodes in the network, the performance decreases in most of the cases. Without the presence of any attack AODV performs good and provided highest PDF. With respect to the increase of number of nodes in the network, the performance getting decreased in most of the cases. But in the case of Black hole attack, with respect to the increase of no of nodes in the network, the performance getting increased because with the high number of nodes, there were chances for developing alternate path that may avoid malicious nodes in it. black hole attack and selfish node attack seems to be providing lower EED AODV(without attack) – but certainly it does not mean that black hole attack and selfish node attack are improving the performance or the network. The low End-to-end delay under this two attacks are due to a strange fact that these two attacks make disconnection in TCP flows and since the packets are not at all forwarded to any further nodes, indirectly it is reduce the message overhead in the network and reduced bandwidth usage otherwise it will be consumed by the forwarded data packets. So, the flows that were unaffected by black hole attack and selfish node attack (the connections where there are no neighboring attack

nodes) utilizes that extra bandwidth and gains some performance in term of some metrics.

2. Periodic Trust based Handshake based AODV (PTH-AODV): The main advantage of the proposed detection and prevention scheme is : it will detect and prevent the malicious nodes in the very early stage of AODV route discovery process. So, it will not need any manipulation in routing tables in the route resolving process, because, by the design, it will avoid including malicious hops in routing table even at the route discovery process itself. In protocols discussed in literature, some of the neighbors those who were silent and not actively participated in communications will get low trust factor and will be wrongly identified as malicious. Because of this wrong identification, the link between source to destination will get broken at different locations on their path because of this false identification of malicious nodes.

In proposed Periodic Trust Handshake based trust AODV (PTH-AODV), it will overcome that problem and reduce the possibility of such false marking of non-malicious nodes as malicious nodes by introducing a Periodic Trust Handshake mechanism.

a) Here the total number of nodes is varied as 40,50 and 60 and keeping the number of malicious node as 15. The plots are taken in terms of total number of sent and received packets, total number of dropped packets, routing load, MAC load, average throughput which determined the protocol efficiency, PDF, EED, Average consumed battery energy and packets dropped at different layers. It was observed that PTH-AODV was able to send and receive as many as packets as normal AODV without any attack. Further, new protocol almost reduces half the number of packets dropped thereby improving the throughput and PDF to a significant amount. However, it hardly causes any changes in EED rather it is a little bit increased in new protocol. Plots show that battery consumption in PTH-AODV is lesser than AODV.

b) PTH-AODV was also compared with already available frAODV and it is found to perform better

3. Dynamic Trust based Handshake based AODV (DTH-AODV): In this work we further developed a dynamic trust handshake based detection of black hole attack. The

performance of the algorithm using a Dynamic Trust Handshake based detection mechanism is further increased. Dynamic Trust Handshake based detection mechanism will detect the malicious nodes very quickly and efficiently in a short term military rescue like MANET scenarios without much increase in overhead. In previously implemented PTH-AODV, trust handshake message will be sent only in periodic intervals but in this proposed DTH-AODV, that was dynamically controlled with respect to the state of the routing process to avoid excess routing overhead of trust messages.

We implemented out DTH-AODV under NS2 and further verification and validation is done by comparing its performance with the results of previous PTH-AODV, AODV and AODV under attack. In terms of sent and received packets DTH-AODV performed a little bit better than previous PTH-AODV. The proposed DTH-AODV improved the throughput and PDF almost equal to that of AODV and performed a little bit better than previously proposed PTH-AODV. EED and battery consumption of PTH-AODV and DTH-AODV is comparable. And routing load of DTH-AODV is less.

The main advantage of the proposed PTH-AODV and DTH-AODV is : they will detect and prevent the malicious nodes in the very early stage of route discovery process. So, they will not need any manipulation in routing tables in the route resolving process, because, by the design, it will avoid including malicious hops in routing table of normal nodes at the route discovery process itself.

In this work, we used unencrypted trust handshake messages in the design of the proposed PTH-AODV and DTH-AODV. This is the main advantage of the protocol that it does not require ant trusted third party infrastructure or encryption methods for its operation. All the feedbacks of the nodes are stored in a separate trust table per node which is modified based on previous node feedback of most active and best route. It also dealt efficiently with selfish attack. Moreover, the most common problem that inactive nodes from a large period are treated as selfish node. This protocol also handles all the cases of selfish node very efficiently.

7.2 Future Scope

1. The proposed protocols PTH-AODV and DTH-AODV are simulated over different conditions of MANET. This is basically a simulation study using NS2 simulation tool. However, the actual results may vary when run on simulation tool. So, the better results can be obtained by implementing it on real world test-bed.

2. The trust based addition can also be added to other routing protocols discussed in section 1.3 to have secure proactive and reactive routing protocols

3. Future works may address the way to apply the trust handshake based method for detecting different kinds of Jellyfish Attacks and other attacks like byzantine attack, Sybil attack etc. of MANET. So that one may address issues related with implenting the detection method for detecting other kinds of attacks.

4. In future works, we may explore more trust dispersal and trust decay strategy over time so as to reduce the trust metric overhead maintenances.

References

[1] Chlamtac, Imrich, Marco Conti, and Jennifer J-N. Liu. "Mobile ad hoc networking: imperatives and challenges." Ad hoc networks 1.1 (2003): 13-64.

[2] Macker, Joseph P., and M. Scott Corson. "Mobile ad hoc networking and the IETF." ACM SIGMOBILE Mobile Computing and Communications Review 2.1 (1998): 9-14.

[3] Varshney, Upkar, and Ron Vetter. "Emerging mobile and wireless networks." Communications of the ACM 43.6 (2000): 73-81.

[4] Davidrajuh, R. "Exploring the use of Bluetooth in Building Wireless Information Systems". International Journal of Mobile Communications (ISSN: 1470-949X) (2005), Vol.5, No.1: 1-10 .

[5] Belding-Royer, E.M. and Toh C.K. "A Review of Current Routing Protocols for Ad-hoc Mobile wireless networks", IEEE Personal Communications Magazine, pp. 46–55, 1999.

[6] Verma, A.K. and Mayank Dave and Joshi, R.C. "Classification of Routing Protocols in MANET",National Symposium on Emerging Trends in Networking & Mobile Communication (NSNM-2003), pp. 132-139, 2003.

[7] Abolhasan, M. And Wysocki, T. And Dutkiewicz, E. "A Review of Routing Protocols for Mobile Ad hoc Networks", Ad Hoc Networks vol. 2(1), pp. 1–22, 2004.

[8] Ahmad, Iftikhar, Uzma Ashraf, and Abdul Ghafoor. "A comparative qos survey of mobile ad hoc network routing protocols." Journal of the Chinese institute of engineers 39.5 (2016): 585-592.

[9] Alubady, Raaid, et al. "Performance analysis of reactive and proactive routing protocols in MANET." ARPN J. Eng. Appl. Sci 10.3 (2015): 1468-1478.

[10] Perkins, Charles E., and Pravin Bhagwat. "Highly dynamic destination-sequenced distance-vector routing (DSDV) for mobile computers." ACM SIGCOMM computer communication review. Vol. 24. No. 4. ACM, 1994.

[11] Dhenakaran, Dr SS, and A. Parvathavarthini. "An overview of routing protocols in mobile ad-hoc network." International Journal of Advanced Research in Computer Science and Software Engineering 3.2 (2013).

[12] Johnson, David B., David A. Maltz, and Josh Broch. "DSR: The dynamic source routing protocol for multi-hop wireless ad hoc networks." Ad hoc networking 5 (2001): 139-172.

[13] Perkins, Charles, Elizabeth Belding-Royer, and Samir Das. Ad hoc on-demand distance vector (AODV) routing. No. RFC 3561. 2003.

[14] Park VD, Corson MS, 1997. A Highly Adaptive Distributed Routing Algorithm for Mobile Wireless Networks. INFOCOM: 97-106.

[15] Chen, Tsu-Wei, and Mario Gerla. "Global state routing: A new routing scheme for ad-hoc wireless networks." Communications, 1998. ICC 98. Conference Record. 1998 IEEE International Conference on. Vol. 1. IEEE, 1998.

[16] Pei, Guangyu, Mario Gerla, and Tsu-Wei Chen. "Fisheye state routing: A routing scheme for ad hoc wireless networks." Communications, 2000. ICC 2000. 2000 IEEE International Conference on. Vol. 1. IEEE, 2000.

[17] Chiang, Ching-Chuan, et al. "Routing in clustered multihop, mobile wireless networks with fading channel." proceedings of IEEE SICON. Vol. 97. No. 1997. 1997.

[18] Jacquet, Philippe, et al. "Optimized link state routing protocol for ad hoc networks." Multi Topic Conference, 2001. IEEE INMIC 2001. Technology for the 21st Century. Proceedings. IEEE International. IEEE, 2001.

[19] Green, David B., and Mohammad S. Obaidat. "Modeling and simulation of IEEE 802.11 WLAN mobile ad hoc networks using topology broadcast reverse-path forwarding (TBRPF)." Computer Communications 26.15 (2003): 1741-1746.

[20] Dhenakaran, Dr SS, and A. Parvathavarthini. "An overview of routing protocols in mobile ad-hoc network." International Journal of Advanced Research in Computer Science and Software Engineering 3.2 (2013).

[21] Bhatia, Tarunpreet, and A. K. Verma. "qos Comparison of MANET Routing Protocols." International Journal of Computer Network and Information Security 7.9 (2015): 64.

[22] Toh, Chai-Keong. "A novel distributed routing protocol to support ad-hoc mobile computing." Computers and Communications, 1996., Conference Proceedings of the 1996 IEEE Fifteenth Annual International Phoenix Conference on. IEEE, 1996.

[23] Dube, Rohit, et al. "Signal stability-based adaptive routing (SSA) for ad hoc mobile networks." IEEE Personal communications 4.1 (1997): 36-45.

[24] Gunes, Mesut, Udo Sorges, and Imed Bouazizi. "ARA-the ant-colony based routing algorithm for MANETs." Parallel Processing Workshops, 2002. Proceedings. International Conference on. IEEE, 2002.

[25] Wang, Jianping, et al. "HOPNET: A hybrid ant colony optimization routing algorithm for mobile ad hoc network." Ad Hoc Networks 7.4 (2009): 690-705.

[26] Singh, Gurpreet, Neeraj Kumar, and Anil Kumar Verma. "Oantalg: An orientation based ant colony algorithm for mobile ad hoc networks." Wireless personal communications 77.3 (2014): 1859-1884.

[27] Robinson, Y. Harold, et al. "TBOR: tree based opportunistic routing for mobile ad hoc networks." World Academy of Science, Engineering and Technology, International Journal of Computer, Electrical, Automation, Control and Information Engineering 10.6 (2016): 1207-1214.

[28] Khan, Nikhat Raza, Sanjay Sharma, and P. S. Patheja. "Energy-Aware Multipath Routing Scheme Based on Particle Swarm Optimization (EMPSO)." Energy 5.01 (2018).

[29] Jiang, M., Li, J., and Tay, Y. C. (1999). Cluster based routing protocol (cbrp) functional specification, internet draft, manet working group.

[30] Pearlman, Marc R., and Zygmunt J. Haas. "Determining the optimal configuration for the zone routing protocol." IEEE Journal on Selected Areas in Communications 17.8 (1999): 1395-1414.

[31] Joa-Ng, Mario, and I-Tai Lu. "A peer-to-peer zone-based two-level link state routing for mobile ad hoc networks." IEEE Journal on selected areas in communications 17.8 (1999): 1415-1425.

[32] Radhakrishnan, Sridhar, et al. "DST-a routing protocol for ad hoc networks using distributed spanning trees." Wireless Communications and Networking Conference, 1999. WCNC. 1999 IEEE. Vol. 3. IEEE, 1999.

[33] Nikaein, Navid, Houda Labiod, and Christian Bonnet. "DDR: distributed dynamic routing algorithm for mobile ad hoc networks." Proceedings of the 1st ACM international symposium on Mobile ad hoc networking & computing. IEEE Press, 2000.

[34] Basagni, Stefano, et al. "A distance routing effect algorithm for mobility (DREAM)." Proceedings of the 4th annual ACM/IEEE international conference on Mobile computing and networking. ACM, 1998.

[35] Ko, Young- Bae, and Nitin H. Vaidya. "Location- Aided Routing (LAR) in mobile ad hoc networks." Wireless networks 6.4 (2000): 307-321.

[36] Rishiwal, Vinay, and Mano Yadav. "Routing in wireless ad hoc networks." International Journal of Internet Protocol Technology 7.2 (2012): 108-119. (TIJCSA) 2.01 (2013).

[37] Mengual Galan, Luis, Liliana Enciso Quispe, and Rommel Torres Tandazo. "Analysis of ad hoc routing protocols for emergency and rescue scenarios." (2012): 781-786.

[38] Torres, Rommel, et al. "A management Ad Hoc networks model for rescue and emergency scenarios." Expert Systems with Applications 39.10 (2012): 9554-9563.

[39] Hogie, Luc, Pascal Bouvry, and Frédéric Guinand. "An overview of MANETs simulation." Electronic notes in theoretical computer science 150.1 (2006): 81-101.

[40] Kumar, Rajeev, Kailash Patidar, and Megha Jain. "A Survey on Routing Protocols with Performance Parameters for Different Number of Nodes." Journal of Network Communications and Emerging Technologies (JNCET) www. Jncet. Org 6.2 (2016).

[41] Choudhury, Prasenjit, Anirban Sarkar, and Narayan C. Debnath. "Deployment of Service Oriented architecture in MANET: A research roadmap." Industrial Informatics (INDIN), 2011 9th IEEE International Conference on. IEEE, 2011.

[42] Salmanian, Mazda, et al. "A modular security architecture for managing security associations in MANETs." Mobile Adhoc and Sensor Systems (MASS), 2010 IEEE 7th International Conference on. IEEE, 2010.

[43] Kumar, Pradeep, and A. K. Vatsa. "Novel security architecture and mechanism for identity based information retrieval system in MANET." Int J Mobil Adhoc Netw 1.3 (2011): 68-72.

[44] Ghalwash, Atef Z., Aliaa AA Youssif, Sherif M. Hashad, and Robin Doss. "Self adjusted security architecture for mobile ad hoc networks (MANETs)." Computer and Information Science, 2007. ICIS 2007. 6th IEEE/ACIS International Conference on. IEEE, 2007.

[45] Bhargavan, Karthikeyan, Davor Obradovic, and Carl A. Gunter. "Formal verification of standards for distance vector routing protocols." Journal of the ACM (JACM) 49.4 (2002): 538-576.

[46] Balakrishnan, Venkat, Vijay Varadharajan, and Uday Tupakula. "Subjective logic based trust model for mobile ad hoc networks." Proceedings of the 4th international conference on Security and privacy in communication netowrks. ACM, 2008.

[47] Wu, Y., Tang, S., Xu, P. And Li, X.Y., 2010. “Dealing with selfishness and moral hazard in noncooperative wireless networks.” IEEE Transactions on Mobile Computing, 9(3), pp.420-434.

[48] Cho, Jin-Hee, Ananthram Swami, and Ray Chen. "A survey on trust management for mobile ad hoc networks." IEEE Communications Surveys & Tutorials 13.4 (2011): 562-583.

[49] D. C. Mishra, Himani Sharma, R. K. Sharma and Naveen Kumar, “A first cryptosystem for security of two dimensional data.” Fractals 25. 01(2017): 1750011.

[50] P. Ning and K. Sun, “How to misuse AODV: A case study of insider attacks against mobile ad-hoc routing protocols,” in Proc. 4th Annu. IEEE Inf. Assurance Workshop, Jun. 2003.

[51] Marti, Sergio, Thomas J. Giuli, Kevin Lai, and Mary Baker. "Mitigating routing misbehaviour in mobile ad hoc networks." Proceedings of the 6th annual international conference on Mobile computing and networking, pp. 255-265. ACM, 2000.

[52] Verma, A.K. and Dave, M. And Joshi, R.C. “Secure Routing in Mobile Networks: A Review,” International J. Of Systemics, Cybernetics and Informatics (IJSCI), vol. 11, pp. 67-74, 2008.

[53] Khan, Muhammad Saleem, et al. "Isolating Misbehaving Nodes in MANETs with an Adaptive Trust Threshold Strategy." Mobile Networks and Applications (2017): 1-17.

[54] Das, Debjit, Koushik Majumder, and Anurag Dasgupta. "Selfish node detection and low cost data transmission in MANET using game theory." Procedia Computer Science 54 (2015): 92-101.

[55] Ngadi, Md. A. And Khokhar, R. H. And Mandala, S. “ A Review Current Routing Attacks in Mobile Ad-hoc Networks”, International Journal of Computer Science and Security, vol. 2 (3). Pp. 18-29. 2008.

[56] Ponsam, J. Godwin, and Dr R. Srinivasan. "A survey on MANET security challenges, attacks and its countermeasures." International Journal of Emerging Trends & Technology in Computer Science (IJETTCS) 3.1 (2014).

[57] Yih-Chun, Hu, and Adrian Perrig. "A survey of secure wireless ad hoc routing." IEEE Security & Privacy 2.3 (2004): 28-39.

[58] Nguyen, Hoang Lan, and Uyen Trang Nguyen. "A study of different types of attacks on multicast in mobile ad hoc networks." Ad Hoc Networks 6.1 (2008): 32-46.

[59] Wu, Bing, Jianmin Chen, Jie Wu, and Mihaela Cardei. "A survey of attacks and countermeasures in mobile ad hoc networks." Wireless network security (2007): 103-135.

[60] Ahamad, Tariq, and Abdullah Aljumah. "Detection and defense mechanism against Dos in MANET." Indian Journal of Science and Technology 8.33 (2015).

[61] Vishnu, K., and Amos J. Paul. "Detection and removal of cooperative black/gray hole attack in mobile ad hoc networks." International Journal of Computer Applications 1.22 (2010): 38-42.

[62] Sen, Jaydip, et al. "A mechanism for detection of gray hole attack in mobile Ad Hoc networks." Information, Communications & Signal Processing, 2007 6th International Conference on. IEEE, 2007.

[63] Tseng, Fan-Hsun, Li-Der Chou, and Han-Chieh Chao. "A survey of black hole attacks in wireless mobile ad hoc networks." Human-centric Computing and Information Sciences 1.1 (2011): 4.

[64] Shandilya, Shishir K., and Sunita Sahu. "A trust based security scheme for RREQ flooding attack in MANET." International journal of computer applications 5.12 (2010): 0975-8887.

[65] Patel, Bipin N., and Tushar S. Patel. "A Survey on Detecting Wormhole Attack in Manet." Journal of Engineering Research and Applications 4.3 (2014): 653-656.

[66] Shastri, Ashka, and Jignesh Joshi. "A Wormhole Attack in Mobile Ad-hoc Network: Detection and Prevention." Proceedings of the Second International Conference on Information and Communication Technology for Competitive Strategies. ACM, 2016.

[67] Laxmi, Vijay, et al. "jellyfish attack: Analysis, detection and countermeasure in TCP-based MANET." Journal of Information Security and Applications 22 (2015): 99-112.

[68] Agrawal, Neha, Krishna Kumar Joshi, and Neelam Joshi. "Performance Evaluation of Byzantine Rushing Attack in ADHOC Network." International Journal of Computer Applications 123.6 (2015).

[69] Hu, Yih-Chun, Adrian Perrig, and David B. Johnson. "Rushing attacks and defense in wireless ad hoc network routing protocols." Proceedings of the 2nd ACM workshop on Wireless security. ACM, 2003.

[70] Chahal, Pooja, Gaurav Kumar Tak, and Anurag Singh Tomar. "Comparative Analysis of Various Attacks on MANET." International Journal of Computer Applications 111.12 (2015).

[71] Navaneethan, T., and M. Lalli. "Security Attacks in Mobile Ad-hoc Networks–A Literature Survey." T. Navaneethan et al, International Journal of Computer Science and Mobile Applications 2.4 (2014): 1-7.

[72] Gupte, Siddhartha, and Mukesh Singhal. "Secure routing in mobile wireless ad hoc networks." Ad Hoc Networks 1.1 (2003): 151-174.

[73] Mamatha, G. S., and S. C. Sharma. "A highly secured approach against attacks in MANETs." International Journal of Computer Theory and Engineering 2.5 (2010): 815.

[74] Jain, Ankit, Arnika Jain, and Pramod Kumar Sagar. "Various security attacks and trust based security architecture for manet." Global Journal of Computer Science and Technology (2010).

[75] Seddik-Ghaleb, Alaa, Yacine Ghamri-Doudane, and Sidi-Mohammed Senouci. "Effect of ad hoc routing protocols on TCP performance within MANETs." Sensor and Ad Hoc Communications and Networks, 2006. SECON'06. 2006 3rd Annual IEEE Communications Society on. Vol. 3. IEEE, 2006.

[76] Aad, Imad, Jean-Pierre Hubaux, and Edward W. Knightly. "Impact of denial of service attacks on ad hoc networks." IEEE/ACM transactions on networking 16.4 (2008): 791-802.

[77] Laxmi, Vijay, et al. "Impact analysis of jellyfish attack on TCP-based mobile ad-hoc networks." Proceedings of the 6th International Conference on Security of Information and Networks. ACM, 2013.

[78] Wazid, Mohammad, Roshan Singh Sachan, and R. H. Goudar. "Measuring the Impact of jellyfish Attack on the Performance of Mobile Ad Hoc Networks using AODV Protocol." Proc. Int. Conf. On Computational Intelligence and Information Technology, CIIT, Elsevier. 2012.

[79] Rajaram, A., and S. Palaniswami. "Malicious node detection system for mobile ad hoc networks." International Journal of Computer Science and Information Technologies 1.2 (2010): 77-85.

[80] Konorski, Jerzy. "Selfishness detection in mobile ad hoc networks: how dissemination of indirect information turns into a strategic issue." Information Technology (ICIT), 2010 2nd International Conference on. IEEE, 2010.

[81] Xia, zhengyou, and Jian Wang. "DIMH: A novel model to detect and isolate malicious hosts for mobile ad hoc network." Computer Standards & Interfaces 28.6 (2006): 660-669.

[82] Kargl, Frank, Andreas Klenk, Stefan Scha lott, and Michael Weber. "Advanced detection of selfish or malicious nodes in ad hoc networks." In ESAS (2004):152-165.

[83] Chauhan, Naveen, Lalit K. Awasthi, Narottam Chand, Ramesh C. Joshi, and Manoj Misra.. "Cooperative Caching in Mobile Ad Hoc Networks." Contemporary Challenges and Solutions for Mobile and Multimedia Technologies. IGI Global, 2013. 255-270.

[84] Al-Shurman, Mohammad, Seong-Moo Yoo, and Seungjin Park. "Black hole attack in mobile ad hoc networks." Proceedings of the 42nd annual Southeast regional conference. ACM, 2004.

[85] Tseng, Fan-Hsun, Li-Der Chou, and Han-Chieh Chao. "A survey of black hole attacks in wireless mobile ad hoc networks." Human-centric Computing and Information Sciences 1.1 (2011): 4.

[86] Jaisankar, N., R. Saravanan, and K. Durai Swamy. "A novel security approach for detecting black hole attack in MANET." Information processing and management (2010): 217-223.

[87] Yu, Chang Wu, et al. "A distributed and cooperative black hole node detection and elimination mechanism for ad hoc networks." Pacific-Asia Conference on Knowledge Discovery and Data Mining. Springer, Berlin, Heidelberg, 2007.

[88] Mohanapriya, M., and Ilango Krishnamurthi. "Modified DSR protocol for detection and removal of selective black hole attack in MANET." Computers & Electrical Engineering 40.2 (2014): 530-538.

[89] Mohammad Iftekhar Husain, S. Upadhyaya and M. Chandrasekaran: A Novel Approach for Security and Robustness in Wireless Embedded Systems. Information Systems and Applications, incl. Internet/Web, and HCI, Lecture Notes in Computer Science, Vol. 5287, pp 323-335, 2008.

[90] Kumar, Sunil, and Kamlesh Dutta. "Security issues in mobile ad hoc networks: A survey." Security, Privacy, Trust, and Resource Management in Mobile and Wireless Communications (2014): 176-221.

[91] Singh, Tejpreet, Jaswinder Singh, and Sandeep Sharma. "Survey of secure routing protocols in MANET." International Journal of Mobile Network Design and Innovation 6.3 (2016): 142-155.

[92] Wang, Dongbin, Mingzeng Hu, and Hui Zhi. "A survey of secure routing in ad hoc networks." Web-Age Information Management, 2008. WAIM'08. The Ninth International Conference on. IEEE, 2008.

[93] Wu, Bing, et al. "Secure and efficient key management in mobile ad hoc networks." Journal of Network and Computer Applications 30.3 (2007): 937-954.

[94] Yang, Hao, et al. "Security in mobile ad hoc networks: challenges and solutions." IEEE wireless communications 11.1 (2004): 38-47.

[95] Abusalah, Loay, Ashfaq Khokhar, and Mohsen Guizani. "A survey of secure mobile ad hoc routing protocols." IEEE communications surveys & tutorials 10.4 (2008).

[96] Sanzgiri, Kimaya, et al. "A secure routing protocol for ad hoc networks." Network Protocols, 2002. Proceedings. 10th IEEE International Conference on. IEEE, 2002.

[97] Papadimitratos, Panagiotis, and Zygmunt J. Haas. "Secure routing for mobile ad hoc networks." SCS Communication Networks and Distributed Systems Modeling and Simulation Conference (CNDS 2002). Vol. 31. 2002.

[98] Hu, Yih-Chun, David B. Johnson, and Adrian Perrig. "SEAD: Secure efficient distance vector routing for mobile wireless ad hoc networks." Ad hoc networks 1.1 (2004): 175-192.

[99] Hu, Yih-Chun, Adrian Perrig, and David B. Johnson. "Ariadne: A secure on-demand routing protocol for ad hoc networks." Wireless networks 11.1-2 (2005): 21-38.

[100] Zapata, Manel Guerrero. "Secure ad hoc on-demand distance vector routing." ACM SIGMOBILE Mobile Computing and Communications Review 6.3 (2002): 106-107.

[101] Papadimitratos, Panagiotis, and Zygmunt J. Haas. "Secure link state routing for mobile ad hoc networks." Applications and the Internet Workshops, 2003. Proceedings. 2003 Symposium on. IEEE, 2003.

[102] Yi, Seung, Prasad Naldurg, and Robin Kravets. "Security-aware ad hoc routing for wireless networks." Proceedings of the 2nd ACM international symposium on Mobile ad hoc networking & computing. ACM, 2001.

[103] R. Ramanujan, A. Ahamad, and K. Thurber, “Techniques for Intrusion Resistant Ad hoc Routing Algorithms (TIARA),” Proc. Military Communications Conf. (MILCOM 2000), Los Angeles, CA, October 2000, pp. 660-664.

[104] S. Capkun, and J.-P. Hubaux, “BISS: Building Secure Routing out of an Incomplete Set of Security Associations,” Proc. ACM Workshop on Wireless Security, ACM Press, 2003, pp. 21-29.

[105] Y.-C. Hu, A. Perrig, and D.B. Johnson, “Packet Leashes: A Defense Against Wormhole Attacks in Wireless Ad hoc Networks,” Proc. 22nd Annual Joint Conf. IEEE Computer and Communications Societies (Infocom’03), San Francisco, CA, April 2003.

[106] A. K. Verma, M. Dave, R. C. Joshi, "Genetic algorithm and tabu search attack on the mono-alphabetic substitution cipher in adhoc networks", Journal of Computer Science, vol. 3, pp. 134-137, 2007.

[107] Sklavos, Nicolas, and Xinmiao Zhang, eds. Wireless security and cryptography: specifications and implementations. CRC press, 2017.

[108] Argyroudis, Patroklos G., and Donal o'mahony. "Secure routing for mobile ad hoc networks." IEEE Communications Surveys and Tutorials 7.1-4 (2005): 2-21.

[109] Ertaul, Levent, and Nitu Chavan. "Elliptic curve cryptography based threshold cryptography (ecc-tc) implementation for MANETs." IJCSNS 7.4 (2007): 48.

[110] Arokiaraj, A. Rex Macedo, and A. Shanmugam. "ACS: An efficient address based cryptography scheme for mobile ad hoc networks security." Computer and Communication Engineering, 2008. ICCCE 2008. International Conference on. IEEE, 2008.

[111] Kim, Jihye, and Gene Tsudik. "SRDP: Secure route discovery for dynamic source routing in MANETs." Ad Hoc Networks 7.6 (2009): 1097-1109.

[112] Nakayama, Hidehisa, et al. "A dynamic anomaly detection scheme for AODV-based mobile ad hoc networks." IEEE transactions on vehicular technology 58.5 (2009): 2471-2481.

[113] Gupta, Rachika. "Mobile Ad hoc Network (MANETs): Proposed solution to Security Related Issues." Indian Journal of Computer Science and Engineering (IJCSE) 2.5 (2011).

[114] Chatterjee, Pushpita, Indranil Sengupta, and Soumya Kanti Ghosh. "STACRP: a secure trusted auction oriented clustering based routing protocol for MANET." Cluster Computing 15.3 (2012): 303-320.

[115] Zhao, Shushan, Robert Kent, and Akshai Aggarwal. "A key management and secure routing integrated framework for mobile ad-hoc networks." Ad Hoc Networks 11.3 (2013): 1046-1061.

[116] Yang, Yang. "Broadcast encryption based non-interactive key distribution in MANETs." Journal of Computer and System Sciences 80.3 (2014): 533-545.

[117] Obaidat, Mohammad S., et al. "A cryptography- based protocol against packet dropping and message tampering attacks on mobile ad hoc networks." Security and Communication Networks 7.2 (2014): 376-384.

[118] Babu, E. Suresh, C. Nagaraju, and MHM Krishna Prasad. "Analysis of Secure Routing Protocol for Wireless Adhoc Networks Using Efficient DNA Based Cryptographic Mechanism." Procedia Computer Science 70 (2015): 341-347.

[119] Ravilla, Dilli, and Chandra Shekar Reddy Putta. "Enhancing the Security of MANETs Using Hash Algorithms." Procedia Computer Science 54 (2015): 196-206.

[120] Mehr, Kamal Adli, and Javad Musevi Niya. "Securing Mobile Ad Hoc Networks Using Enhanced Identity- Based Cryptography." ETRI Journal 37.3 (2015): 512-522.

[121] Moudgil, Suveg, and Sanjeev Rana. "A Secure & Robust Scheme to Isolate DDoS Attacks Over MANET." International Journal of Computer Science Issues (IJCSI) 13.3 (2016): 31.

[122] Sharma, Dhruvi, Vimal Kumar, and Rakesh Kumar. "Prevention of wormhole attack using identity based signature scheme in MANET." Computational Intelligence in Data Mining—Volume 2. Springer, New Delhi, 2016. 475-485.

[123] Umar, Muhammad Muneer, Amjad Mehmood, and Houbing Song. "SeCRoP: secure cluster head centered multi- hop routing protocol for mobile ad hoc networks." Security and Communication Networks 9.16 (2016): 3378-3387.

[124] Tan, Shuaishuai, Xiaoping Li, and Qingkuan Dong. "TrustR: An integrated router security framework for protecting computer networks." IEEE Communications Letters 20.2 (2016): 376-379.

[125] Subbulakshmi, P., and S. Vimal. "Secure data packet transmission in manet using enhanced identity-based cryptography (EIBC)." International Journal of New Technologies in Science and Engineering 3.12 (2016): 35-42.

[126] Amir, Mohammad, Dhanroop Mal Nagar, and Vinay Baghela. "Secure DSR Routing Protocol Based on Homomorphic Digital Signature." Proceedings of the International Conference on Advances in Information Communication Technology & Computing. ACM, 2016.

[127] Dorri, Ali. "An EDRI-based approach for detecting and eliminating cooperative black hole nodes in MANET." Wireless Networks 23.6 (2017): 1767-1778.

[128] Rajkumar, Banoth, and Gugua lothu Narsimha. "Secure Light Weight Encryption Protocol for MANET." International Journal of Intelligent Engineering and Systems 10.3 (2017): 58-65.

[129] Mahmood, Khalid, et al. "An elliptic curve cryptography based lightweight authentication scheme for smart grid communication." Future Generation Computer Systems 81 (2018): 557-565.

[130] Marias, Giannis F., et al. "Cooperation enforcement schemes for MANETs: A survey." Wireless Communications and Mobile Computing 6.3 (2006): 319-332.

[131] Mandalas, K., D. Flitzanis, G. F. Marias, and P. Georgiadis. "A survey of several cooperation enforcement schemes for MANETs." In Signal Processing and Information

Technology, 2005. Proceedings of the Fifth IEEE International Symposium on, pp. 466-471. IEEE, 2005.

[132] Wang, Yao, and Julita Vassileva. "Toward trust and reputation based web service selection: A survey." International Transactions on Systems Science and Applications 3.2 (2007): 118-132.

[133] Selvaraj, Chithra, and Sheila Anand. "A survey on security issues of reputation management systems for peer-to-peer networks." Computer Science Review 6.4 (2012): 145-160.

[134] Yu, Yao, Lei Guo, Xingwei Wang, and Cuixiang Liu. "Routing security scheme based on reputation evaluation in hierarchical ad hoc networks." Computer Networks 54, no. 9 (2010): 1460-1469.

[135] Senthilkumar, S., and J. William. "A survey on reputation based selfish node detection techniques in mobile ad hoc network." Journal of theoretical & applied information technology 60.2 (2014).

[136] Manikandan, S. P., and R. Manimegalai. "Survey on mobile Ad Hoc network attacks and mitigation using routing protocols." American Journal of Applied Sciences 9.11 (2012): 1796.

[137] Yoo, Younghwan, and Dharma P. Agrawal. "Why does it pay to be selfish in a MANET?." IEEE Wireless Communications 13.6 (2006): 87-97.

[138] Janzadeh, Hamed, et al. "A secure credit-based cooperation stimulating mechanism for MANETs using hash chains." Future Generation Computer Systems 25.8 (2009): 926-934.

[139] Hoffman, Kevin, David Zage, and Cristina Nita-Rotaru. "A survey of attack and defense techniques for reputation systems." ACM Computing Surveys (CSUR) 42.1 (2009): 1.

[140] S. Marti, T.J. Giuli, K. Lai, and M. Baker, “Mitigating Routing Misbehaviour in Mobile Ad hoc Networks,” Proc. 6th Annual ACM/IEEE Int’l. Conf. Mobile Computing and Networking (Mobicom’00), Boston, Massachusetts, August 2000, pp. 255-265.

[141] Buchegger, Sonja. "Performance Analysis of the CONFIDANT Protocol: Cooperation Of Nodes-Fairness In Distributed Ad-hoc netwoks." mobihoc 2002, Lausanne (2002).

[142] Michiardi, Pietro, and Refik Molva. "Core: a collaborative reputation mechanism to enforce node cooperation in mobile ad hoc networks." Advanced communications and multimedia security. Springer US, 2002. 107-121.

[143] Eschenauer, Laurent, Virgil D. Gligor, and John Baras. "On trust establishment in mobile ad-hoc networks." International Workshop on Security Protocols. Springer, Berlin, Heidelberg, 2002.

[144] Cho, Jin-Hee, Ananthram Swami, and Ray Chen. "A survey on trust management for mobile ad hoc networks." IEEE Communications Surveys & Tutorials 13.4 (2011): 562-583.

[145] Vijayan, R., and N. Jeyanthi. "A survey of trust management in mobile ad hoc networks." International Journal of Applied Engineering Research 11.4 (2016): 2833-2838.

[146] Govindan, Kannan, and Prasant Mohapatra. "Trust computations and trust dynamics in mobile adhoc networks: A survey." IEEE Communications Surveys & Tutorials 14.2 (2012): 279-298.

[147] Yan, Zheng, Peng Zhang, and Athanasios V. Vasilakos. "A survey on trust management for Internet of Things." Journal of network and computer applications 42 (2014): 120-134.

[148] Mishra, Amitabh, and Ketan M. Nadkarni. "Security in wireless ad hoc networks." The handbook of ad hoc wireless networks. CRC Press, Inc., 2003.

[149] Virendra, Mohit, et al. "Quantifying trust in mobile ad-hoc networks." Integration of Knowledge Intensive Multi-Agent Systems, 2005. International Conference on. IEEE, 2005.

[150] Ahmed, Adnan, Kamalrulnizam Abu Bakar, Muhammad Ibrahim Channa, Khalid Haseeb, and Abdul Waheed Khan. "A survey on trust based detection and isolation of malicious nodes in ad-hoc and sensor networks." Frontiers of Computer Science 9, no. 2 (2015): 280-296.

[151] Cho, Jin-Hee, Ananthram Swami, and Ray Chen. "A survey on trust management for mobile ad hoc networks." IEEE Communications Surveys & Tutorials 13.4 (2011): 562-583.

[152] Abusalah, Loay, Ashfaq Khokhar, and Mohsen Guizani. "A survey of secure mobile ad hoc routing protocols." IEEE communications surveys & tutorials 10.4 (2008).

[153] Kukreja, Deepika, Sanjay Kumar Dhurandher, and B. V. R. Reddy. "Enhancing the security of dynamic source routing protocol using energy aware and distributed trust mechanism in MANETs." Intelligent distributed computing. Springer, Cham, 2015. 83-94.

[154] A. A. Pirzada and C. Mcdonald, “Trust Establishment In Pure Ad-hoc Networks,” Wireless Personal Communications, Springer, pp. 139–163, 2006.

[155] Sun, Yan Lindsay, et al. "Information theoretic framework of trust modeling and evaluation for ad hoc networks." IEEE Journal on Selected Areas in Communications 24.2 (2006): 305-317.

[156] Abusalah, Loay, A. Khokhar, G. Benbrahim, and W. Elhajj. "TARP: trust-aware routing protocol." In Proceedings of the 2006 international conference on Wireless communications and mobile computing, pp. 135-140. ACM, 2006.

[157] Balakrishnan, Venkat, Vijay Varadharajan, Udaya Kiran Tupakula, and Phillip Lucs. "Trust and recommendations in mobile ad hoc networks." In Networking and Services, 2007. ICNS. Third International Conference on, pp. 64-64. IEEE, 2007.

[158] Moe, Marie EG, Bjarne E. Helvik, and Svein J. Knapskog. "TSR: Trust-based secure MANET routing using hmms." Proceedings of the 4th ACM symposium on qos and security for wireless and mobile networks. ACM, 2008.

[159] Eissa, Tameem, Shukor Abdul Razak, Rashid Hafeez Khokhar, and Normalia Samian. "Trust-based routing mechanism in MANET: Design and implementation." Mobile Networks and Applications 18, no. 5 (2013): 666-677.

[160] Mohanapriya, M., and Ilango Krishnamurthi. "Trust based DSR routing protocol for mitigating cooperative black hole attacks in ad hoc networks." Arabian Journal for Science and Engineering 39.3 (2014): 1825-1833.

[161] Abdel-Halim, Islam Tharwat, Hossam Mahmoud Ahmed Fahmy, and Ayman Mohammad Bahaa-Eldin. "Agent-based trusted on-demand routing protocol for mobile ad-hoc networks." Wireless Networks 21.2 (2015): 467-483.

[162] Bhargavi, V. Sesha, and S. Viswanadha Raju. "Enhancing security in MANETs through trust-aware routing." Wireless Communications, Signal Processing and Networking (wispnet), International Conference on. IEEE, 2016.

[163] Khan, Muhammad Saleem, Daniele Midi, Majid I. Khan, Nadeem Javaid, and Elisa Bertino. "Isolating Misbehaving Nodes in MANETs with an Adaptive Trust Threshold Strategy." Mobile Networks and Applications 22, no. 3 (2017): 493-509.

[164] Sun, Boyuan, and Donghui Li. "A Comprehensive Trust-Aware Routing Protocol With Multi-Attributes for wsns." IEEE Access 6 (2018): 4725-4741.

[165] T. Ghosh, N. Pissinou, and K. Makki, “Collaborative Trust-based Secure Routing Against Colluding Malicious Nodes in Multi-hop Ad Hoc Networks,” in Proc. 29th Annual IEEE International Conference on Local Computer Networks, pp. 224-231, 2004.

[166] T. Ghosh, N. Pissinou, and K. Makki, “Towards Designing a Trusted Routing Solution in Mobile Ad Hoc Networks,” Mobile Networks and Applications, Springer Science, vol. 10, pp. 985-995, 2005.

[167] A. A. Pirzada, A. Datta, C. Mcdonald, “Trust-based routing for ad-hoc wireless networks,” IEEE, pp. 326-30, 2004.

[168] X. Li, M. R. Lyu, and J. Liu, “A Trust Model Based Routing Protocol for Secure Ad Hoc Networks,” in Proc. Aerospace Conference, IEEE, vol. 2, pp. 1286-1295, 2004.

[169] Rahman, Muhammad Azizur, Algirdas Pakštas, and Frank Zhigang Wang. "Network modelling and simulation tools." Simulation Modelling Practice and Theory 17.6 (2009): 1011-1031.

[170] Camp, Tracy, Jeff Boleng, and Vanessa Davies. "A survey of mobility models for ad hoc network research." Wireless communications and mobile computing 2.5 (2002): 483-502.

[171] Issariyakul, Teerawat, and Ekram Hossain. Introduction to network simulator NS2. Springer Science & Business Media, 2011.

[172] Chhabra, Jitender Kumar. "Improving package structure of object-oriented software using multi-objective optimization and weighted class connections." Journal of King Saud University-Computer and Information Sciences (2015).

[173] Fall, K., and K. Varadhan. "The ns manual. Notes and documentation on the software NS2-simulator, 2002." URL: www. Isi. Edu/nsnam/ns.

[174] Drakos, N., and R. Moore. "NS2-The Manual (formerly Notes and Documentation), 1999."

[175] Mohapatra, S., and P. Kanungo. "Performance analysis of AODV, DSR, OLSR and DSDV routing protocols using NS2 Simulator." Procedia Engineering 30 (2012): 69-76.

[176] Rani, Amita, and Mayank Dave. "Performance evaluation of modified AODV for load balancing 1." Journal of computer science 3 (2007): 863–868.

[177] Li, Bing-Zhao, Ran Tao, and Yue Wang. "New sampling formulae related to linear canonical transform." Signal Processing 87.5 (2007): 983-990.

[178] Quispe, Liliana Enciso, and Luis Mengual Galan. "Behaviour of Ad Hoc routing protocols, analyzed for emergency and rescue scenarios, on a real urban area." Expert systems with applications 41.5 (2014): 2565-2573.