IMPROVED 2ACK SCHEME FOR REDUCING ROUTING OVERHEAD IN MANETs

Dissertation submitted to the Central University of Punjab

For the award of

Master of Technology

In

Centre for Computer Science and Technology

BY

Deepika Dhiman

Supervisor - Er. Neha Sood

Administrative Guide - Prof. (Dr.) A. K. Jain

Centre for Computer Science and Technology

School of Engineering and Technology

Central University of Punjab, Bathinda

September 2014

CERTIFICATE

I declare that the dissertation entitled "IMPROVED 2ACK SCHEME FOR REDUCING ROUTING OVERHEAD IN MANETs" has been prepared by me under the guidance of Er. Neha Sood, Assistant Professor, Centre for Computer Science and Technology, School of Engineering and Technology, Central University of Punjab.

No part of this dissertation/thesis has formed the basis for the award of any degree or fellowship previously.

Deepika Dhiman
Centre for Computer Science and Technology,
School of Engineering and Technology,
Central University of Punjab, Bathinda - 151001.
Date:

CERTIFICATE

I certify that Deepika Dhiman has prepared her dissertation entitled "IMPROVED 2ACK SCHEME FOR REDUCING ROUTING OVERHEAD IN MANETs", for the award of M.Tech. degree of the Central University of Punjab, under my guidance. She has carried out this work at the Centre for Computer Science and Technology, School of Engineering and Technology, Central University of Punjab.

Er. Neha Sood
(Supervisor)
Centre for Computer Science and Technology,
School of Engineering and Technology,
Central University of Punjab, Bathinda - 151001.
Date:

Prof. (Dr.) A. K. Jain
(Administrative Guide)
Centre for Computer Science and Technology,
School of Engineering and Technology,
Central University of Punjab, Bathinda - 151001.
Date:

ABSTRACT

IMPROVED 2ACK SCHEME FOR REDUCING ROUTING OVERHEAD IN MANETs

Name of student: Deepika Dhiman

Registration number: CUPB/MTECH/SET/CST/2012-13/05

Degree for which submitted: Master of Technology

Name of supervisor: Er. Neha Sood

Name of centre: Centre for Computer Science and Technology

Name of school: School of Engineering and Technology

Key words: MANETs, DSR protocol, MAC, routing misbehavior, selfish node, 2ACK scheme.

An autonomous collection of mobile nodes communicating with each other via wireless links either directly or indirectly or relying on other nodes forms the mobile adhoc network (MANET). The routing protocols in MANET are designed based on the assumption that all the participating nodes co-operate. Due to certain issues like open structure and limited energy supply, the nodes sometimes misbehave and act in a selfish manner. 2ACK scheme serves as an add-on technique on few routing protocols (e.g. DSR) to detect such misbehaviour and to mitigate their adverse effect. The major limitation of this 2ACK scheme is additional routing overhead due to authenticated 2ACK packets. Thus, this dissertation work focuses on reducing the routing overhead and end-to-end delay by modifying the authentication mechanism in existing 2ACK scheme. Analytical and simulation results have been presented for evaluating the performance of the work done.

Deepika Dhiman	Er. Neha Sood	Prof. (Dr.) A. K. Jain
	(Supervisor)	(Administrative Guide)

ACKNOWLEDGEMENTS

Words often fall short when we have to express gratitude to the ones who extend their immense support in the hour of need. I bow to the Almighty for providing me strength for timely completion of this work with sincerity and dedication. I would like to thank my advisor **Er. Neha Sood** for her continuous guidance, timely devotion, professional insight and instructions. I extend my heartfelt gratitude to my peers – Shifali Hans and Bandana Bhatia for their help and co-operation in the completion of this work. I also extend my sincere appreciation to Prof. Dr. A. K. Jain, COC, Centre for Computer Science & Technology and the staff of Computer Science & Technology department for providing such an academic environment. Last but not the least, I am thankful to my family for their emotional support, love, blessings and inspiration.

Deepika Dhiman

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF APPENDICES

LIST OF ABBREVIATIONS

Sr. No.	Full Form	Abbreviation
1.	Adhoc On-demand Distance Vector	AODV
2.	Adhoc On-demand Multipath Distance Vector Routing	AOMDV
3.	Authenticated Routing for Adhoc Network	ARAN
4.	Co-operation Of Nodes - Fairness In Dynamic Adhoc NeTwork	CONFIDANT
5.	COllaborative REputation	CORE
6.	Destination Sequenced Distance Vector	DSDV
7.	Dynamic State Routing	DSR
8.	Internet Engineering Task Force - Request For Comments	IETF RFC
9.	Low Overhead Truthful Routing Protocol	LOTTO
10.	Mobile Adhoc NETwork	MANET
11.	End-to-end ACKnowledgement	NACK
12.	Network Simulator	NS2
13.	Observation based Co-operation Enforcement in Ad hoc networks	OCEAN
14.	Optimized Link State Routing Protocol	OLSR
15.	Selective ACKnowledgement	SACK
16.	Self Centered Friendship Tree	SCF
17.	Secure and Objective Reputation based Incentive System	SORI
18.	Secure Routing Protocol	SRP
19.	Secure Trusted Auction Oriented Clustering Based Routing Protocol	STACRP
20.	Selective Two ACKnowledgement	S-TWOACK

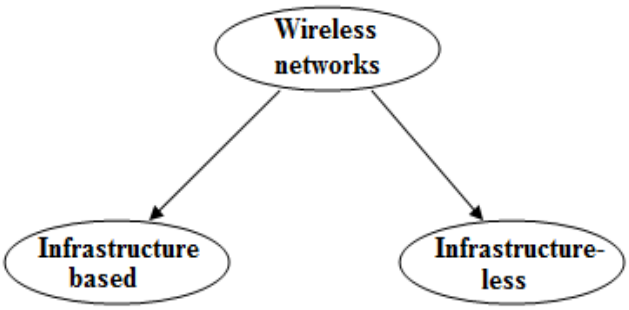
21.	Truthful Multipath Routing Protocol	TMRP
22.	Temporally-Ordered Routing Algorithm	TORA
23.	Two ACKnowledgement	TWO-ACK
24.	2ACKnowledgement	2ACK

CHAPTER 1

INTRODUCTION

Internet usage is skyrocketing day by day with the increasing number of mobile and ubiquitous access. With the proliferation of mobile devices such as cell phones, laptops, palm tops etc., the demand for continuous network connectivity of the physical location has spurred interest in mobile networks for the following reasons-mobility, reduced installation time, increased reliability and long-term cost savings. Wireless and mobile computing allows for spontaneous networking with or without prior set-up of an infrastructure (Mohapatra & Krishnamurthy, 2005) (Sarkar, Basavaraju, & Puttamadappa, 2007).

Wireless networks have been categorized into following two classes:



- Infrastructure based networks

  These networks rely on fixed, wired gateways- access points, base stations. A mobile unit within this network connects to the nearest base station. As the mobile travels from the range of one base station into the other base station's range, a "handoff" takes place from the old base station to the new. Hence, the mobile is able to continue communication seamlessly throughout the network. Wireless local area network is the typical application of infrastructure based network (Ismail & Jaafar, 2007).

- Infrastructure-less networks

  These networks have no fixed routers so any node can perform the functionality of the router, thus discovering and maintaining routes to the other nodes in the network. All the nodes are capable of dynamic and arbitrary movement (Mohapatra & Krishnamurthy, 2005). These networks are basically used in emergency search-and-rescue operations, meetings

or conventions in which persons wish to quickly share information and data acquisition operations in inhospitable terrains, rare animal tracking, space exploration, under sea operations (Ismail & Jaafar, 2007). This type of network is even referred as an “ad hoc” network. The Latin term “ad hoc” means “for this purpose”. Hence, ad hoc network is used to connect wireless clients directly together, without the need for a wireless access point or a connection to the existing wired network (Boukerche, 2008).

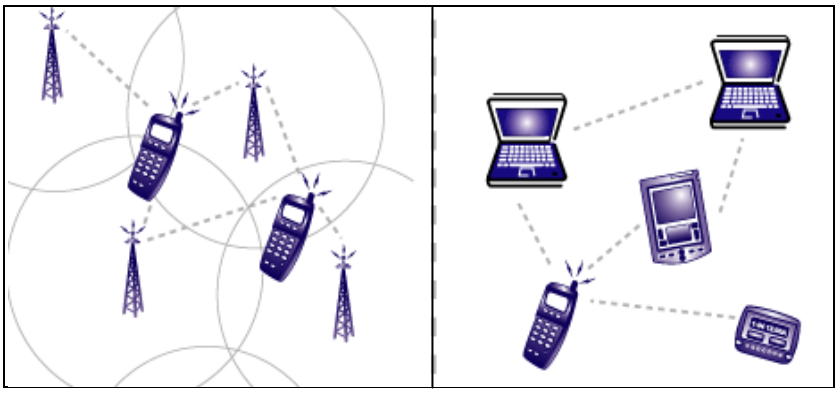


Figure 1: Infrastructure Network (Left) v/s Infrastructure-less Network (Right)

A “mobile ad hoc network”, type of infrastructure-less network, is referred as the collection of wireless mobile nodes forming a temporary network without the help of any established infrastructure or centralized administration (K. Balakrishnan, J. Deng, & V. Varshney, 2005). In such an environment, it becomes mandatory for a single mobile host to enlist the help of other hosts in forwarding the packet to its destination, due to the defined and limited range of each mobile host’s wireless transmissions. Network nodes in MANET are free to move randomly (Jhaveri, Patel, & Jinwala, 2012) (Wu, Chen, Wu, & Cardei, 2007).

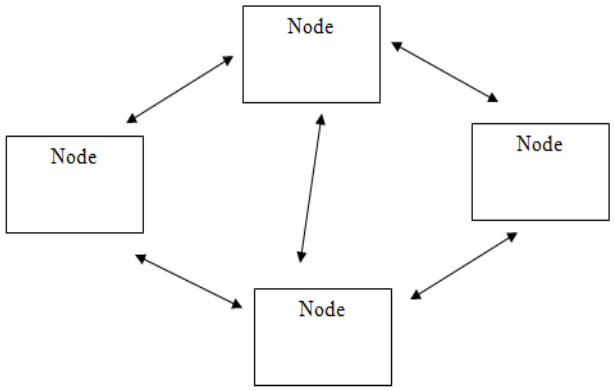


Figure 2: Mobile Adhoc Network

According to IETF RFC 2501, MANET has following characteristics (Basagni, Conti, Giordano, & Stojmenovic, 2004) (Sarkar *et al.*, 2007):

- The nodes can move freely, hence allowing the spontaneous formation and deformation of the mobile network having symmetric or asymmetric links. It is even referred as the self organizing and adapting network.
- The mobile nodes are resource constrained in terms of battery power, bandwidth and energy consumption. For example- Ad hoc nodes like laptops, PDAs etc. have limited power supply and can't generate power of their own.
- Nodes communicate wirelessly and share the same media (radio, infrared)
- Each mobile node supports a distributed peer-to-peer mode, hence acting as an independent router. Thus, it results in the formation of multi-hop network also.
- The capacity of the wireless link in MANET is small, bandwidth constrained and variable in nature with susceptibility to external noise, interference and signal attenuation effects.

On the basis of the area of application, MANET can be classified as (Jain & Tokekar, 2011):

- Closed MANET

  All the nodes in the network co-operate with each other towards a common desired goal. A new node can join only after having permission from the authorized nodes already in the network. Military application is an example of closed MANET.

- Open MANET

  Each node in the network has its own operational goals. The nodes are free to join and leave the network. Virtual classroom session is an example of open MANET.

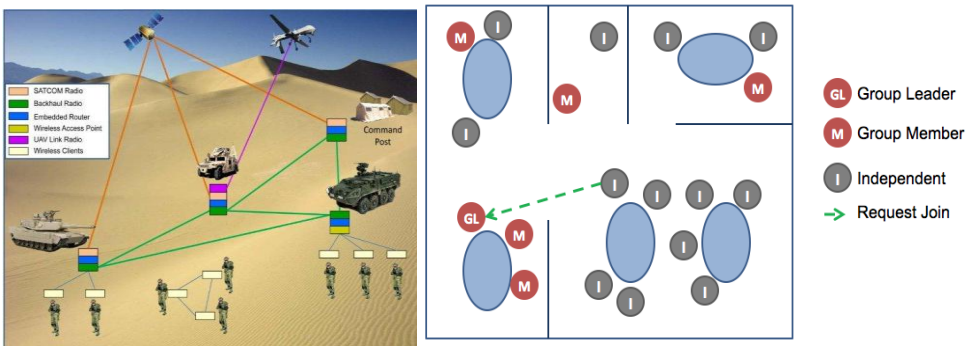


Figure 3: Closed and Open MANET

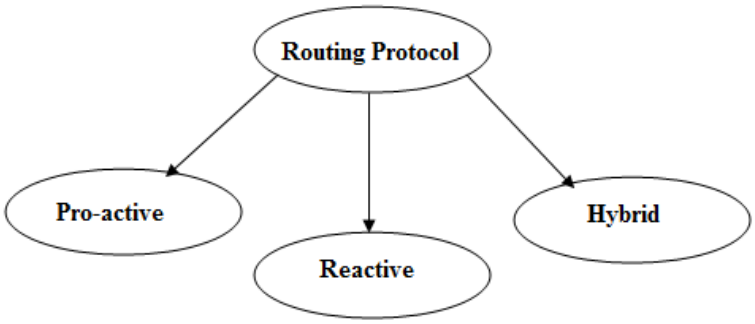
Earlier, MANETs were used in military operations but now it is providing services in other fields also. The following table gives an overview of the MANET applications.

Table 1: Mobile Adhoc Network Applications (Basagni *et al.*, 2004)

Applications	**Possible service of Ad Hoc Networks**
Tactical networks	Military communication, military operations, in the battlefield
Emergency services	Search and rescue operations, replacement of fixed infrastructure in environmental disasters, fire fighting
Coverage extension	Extending cellular network access, linking up with the internet, intranet
Sensor networks	Smart sensors and actuators embedded in consumer electronics, data tracking of environmental conditions, chemical and biological detection
Education	Classrooms, campus settings, meetings or lectures
Entertainment	Multi user games, robotic pets, theme parks, outdoor internet access
Home and enterprise	Conferences, meeting rooms, wireless networking in home or office
Context aware services	• Follow on services: call forwarding, mobile workspace • Information services: location specific services, time dependent services • Infotainment: tourist information
Commercial and civilian environment	• E-commerce: electronic payments • Business: dynamic data access, mobile offers • Vehicular services: road or accident guidance, transmission of road and weather conditions, taxi cab network, inter-vehicle network • Sports stadium, trade fairs, shopping malls

The highly dynamic nature of MANET results in frequent unpredictable changes in network topologies, thus leading to difficulty and complexity to routing among the mobile nodes. The main aim of an ad-hoc network routing protocol is to establish

correct and efficient route between a pair of nodes for reliable and timely delivery of the messages (Basagni *et al.*, 2004). The routing protocols in MANET have been categorized as:



- Proactive protocols

  These routing protocols work to maintain consistent, up-to-date routing information between each and every pair of the nodes in the network by propagating periodic and event driven messages (Basagni *et al.*, 2004) (Kannhavong, Nakayama, Nemoto, Kato, & Jamalipour, 2007). These pro-active routing protocols are even referred as “table driven” protocols because the resulting routing information is maintained in tables. The route discovery process needs large information with high bandwidth and slow reaction. Examples of proactive protocols involves Destination-Sequenced Distance-Vector Protocol, Cluster-head Gateway Switch Routing protocol, Wireless Routing Protocol, Global State Routing, Optimized Link State Routing Protocol, Fisheye State Routing Protocol, Landmark Routing Protocol and Hierarchical State Routing (Jhaveri *et al.*, 2012).

- Reactive protocols

  Every mobile node in the network tries to obtain a route to the destination in a demand fashion, i.e. the route between two nodes is discovered only when it is needed. These protocols do not maintain information related to up-to-date route to any destination in the network and do not even exchange any periodic control messages (Mohapatra & Krishnamurthy, 2005). These protocols tend to reduce overhead with the minimum requirement of bandwidth but they require large amount of time for route discovery resulting in flooding and congestion problems. Ad Hoc On Demand Distance Vector, Dynamic Source Routing, Temporally Ordered

Routing Algorithm, Cluster Based Routing Protocol, Location Aided Routing and Ant Colony Based Routing Algorithm are the on-demand protocols (Sarkar *et al.*, 2007).

- Hybrid routing protocols

  The combined features of both the table-driven and source-initiated on-demand driven protocols are embedded in hybrid protocols. These protocols are basically designed to increase scalability by allowing the nodes with close proximity to work together to form some sort of the backbone for reducing the routing overhead and maximizing the efficiency (Ilyas, 2002). This can be obtained by proactively maintaining the routes to the nearby nodes and then determining the routes to far away nodes using a route discovery strategy. Most of the hybrid protocols are zone based, i.e. the network is partitioned into a number of zones by each node while other group nodes are formed into trees or clusters. Every mobile node acts reactively in the region close to its proximity and acts proactively outside of the region or zone defined (Boukerche, 2008). Zone Routing Protocol, Zone-Based Hierarchical Link State, Scalable Location Updates Routing Protocol, Distributed Spanning Trees Based Routing Protocol and Distributed Dynamic Routing Protocol are some of the hybrid protocols (Sarkar *et al.*, 2007).

1.1. Introduction to Routing Protocol - DSR

DSR is the simplest and efficient routing protocol designed especially for the use in the wireless multi-hop ad hoc network of the mobile nodes. It allows the network to self-organize and self-configure without the requirement of any existing network infrastructure (Maltz & Broch, 2001).

The DSR protocol contains two mechanisms -

- Route Discovery mechanism is the process by which the source node S ‘that wishes to send the data packet to a destination node D will obtain the source route to D. This mechanism is used only when source node attempts to send a packet to destination while it is unaware of the route to destination node (Maltz & Broch, 2001).
- Route Maintenance mechanism is the process by which source node is able to detect while using the source route to destination node, incase the network

topology changes such that it no longer makes use of its route to the destination node. When the route maintenance process indicates that the source route is broken, source node makes an attempt to use any other route to the destination node or it invokes the route discovery process again for finding a new route. This process is used only when source node is actually sending packets to destination node (Maltz & Broch, 2001).

Route Discovery and **Route Maintenance** processes operate entirely on demand. The use of source routing allows the packet routing to be trivially loop-free, hence avoids the need for the maintenance of up-to-date routing information in the intermediate nodes through which the packets are being forwarded and also allows the other nodes that are forwarding or overhearing the data packets to cache the routing related information for further use. All the aspects of DSR protocol operate entirely on-demand, thereby allows the routing packet overhead of DSR to scale automatically to only that needed for reacting to the route changes currently in use (Johnson, Maltz, & Broch, 2001).

Route Discovery

When a source node say S originates a new data packet destined for another destination node D, it places a source route in the header of the packet giving the sequence of the hops that the data packet should follow on its way to node D. Normally, the node S will obtain a particular source route by searching for the route cache previously learned, but if there is no route found in its cache, then it will begin the route discovery process to dynamically find a new route to the node D. In this case, we call node S as the initiator and node D as the target (Maltz & Broch, 2001).

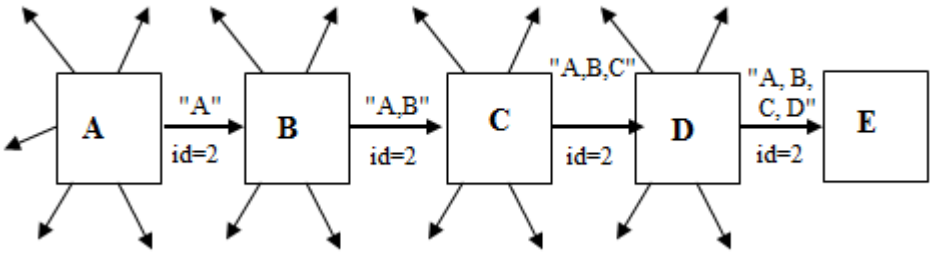


Figure 4: Route Discovery Process in DSR Protocol

In the above figure, a node A attempts to discover a route to node E. To initiate the Route Discovery, A transmits a ROUTE REQUEST message which is received by all the nodes presently within the wireless transmission range of A. Each ROUTE

REQUEST message first identifies the initiator and the target of the route discovery process. Each ROUTE REQUEST contains a unique request id which is determined by the initiator of the REQUEST and the record having list of the address of each and every intermediate node through which the specified copy of the ROUTE REQUEST message has been forwarded. The initiator then initializes the route record to an empty list in the route discovery process. When another node say target of the route discovery receives the ROUTE REQUEST, it returns the ROUTE REPLY message to the route discovery process's initiator giving the copy of the accumulated record of the route from the ROUTE REQUEST. When the initiator will receive this ROUTE REPLY, it first caches this route in the route cache for sending the subsequent packets to the destination. If the node receiving the ROUTE REQUEST finds another ROUTE REQUEST message from this initiator having the same request id or if it finds that its own address has already been listed in the route record of the ROUTE REQUEST message, then it discards the REQUEST. While returning the ROUTE REPLY message to the initiator of the route discovery process such as node E replying to node A, then node E will examine its route cache for the route back to node A for the delivery of the packet containing ROUTE REPLY. Else, node E will perform its own route discovery process for the target node A but to avoid indefinite recursion of route discoveries, it will piggyback this ROUTE REPLY on its own ROUTE REQUEST message for node A (Maltz & Broch, 2001).

Route Maintenance

When forwarding the data packet using a source route, each mobile node that transmits the data packet is responsible for providing the confirmation that the packet has been successfully received by the next hop along that source route. The data packet is retransmitted again and again until this confirmation of receipt is received (Maltz & Broch, 2001).

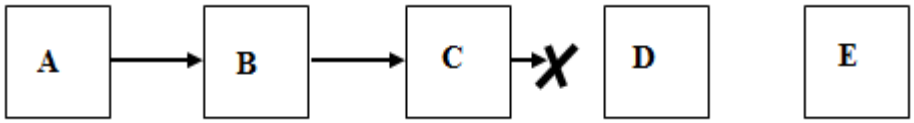


Figure 5: Route Maintenance Process in DSR Protocol

In figure 5, the node A originates a data packet for the node E by the use of a source route through the intermediate nodes B, C and D. In this case, the node A is responsible for receipt of the data packet at node B which is responsible for the

receipt of data packets at node C. Further, the node C is responsible for receipt of data packets at the node D which is responsible for receipt finally at the destination node marked as E. If the data packet is being retransmitted by some hop a maximum number of times and no receipt for the confirmation is received, then this node sends a ROUTE ERROR message towards the original sender of the data packet and identifies the link over which the data packet could not be forwarded. As shown in the figure 5, the data packets from the node C are not delivered to the next hop D, then the node C will return a ROUTE ERROR message to the node A stating that the link starting from node C towards node D is presently “broken.” Then, the node A will remove the broken link from its cache table. Any retransmission of the original packet is a function for upper layer protocols such as TCP. For sending such retransmission or other data packets to the same destination node E, if node A contains in its route cache some other route to node E, then it can send the data packet using the new route instantly. Otherwise, it can even start a new route discovery for this defined target (Maltz & Broch, 2001).

Typical features of MANET contributing to the vulnerabilities (Mohapatra & Krishnamurthy, 2005) (Goyal, Parmar, & Rishi, 2011):

- Unreliability of the wireless links between the nodes

  Due to the mobility of the nodes and the limited energy supplied to them, the wireless links between these nodes in the ad hoc network are not consistent for the communicating parties.
- Dynamic topology

  Due to the continuous movement of the nodes, the topology of the ad hoc network keeps on changing continuously. The nodes can randomly move in and out of the radio range of the other nodes in the mobile ad hoc network, and hence the routing information will be fluctuating.
- Lack of incorporation of the security features in the wireless routing protocol

  The topology of the mobile ad hoc networks is dynamic in nature, hence it becomes necessary for every pair of the adjacent nodes to incorporate the routing issue in order to prevent potential attacks that make use of the vulnerabilities in the statically configured routing protocol.

Hence, the mobile ad hoc networks are prone to suffer from the malicious behaviors such as flooding attack, black hole attack, rushing attack. Therefore, we need to pay more attention to the security issues in the mobile ad hoc networks. A variety of attacks like modification, fabrication, sybil attack, wormhole attack (tunneling), denial of service attack, black hole attack, invisible node attack, rushing attack and **"non-cooperation"** tend to reduce the reliability of these routing protocols (Goyal *et al.*, 2011). To overcome these security threats, the concept of secured routing protocols came into existence. Some popular secured routing algorithms are SRP, ARAN, Ariadne, SEAD etc. Whereas most of the attacks that are based on the manipulations of the routing data can be detected using some secure routing protocol like ARAN, Ariadne but when a node misbehaves, all of the secure routing protocols fail.

A node may become selfish or misbehaving due to honest as well as malicious reasons. Honest reasons involve collisions, channel error, buffer overflow etc. Malicious reasons could be due to attacks (black hole attack or wormhole attack) on the node, congestion etc (Chatterjee, Sengupta, & Ghosh, 2012). Hence, a selfish node tries to save its own energy and bandwidth, minimizes packet transfer and maximizes the packet delivery time and packet loss rate.

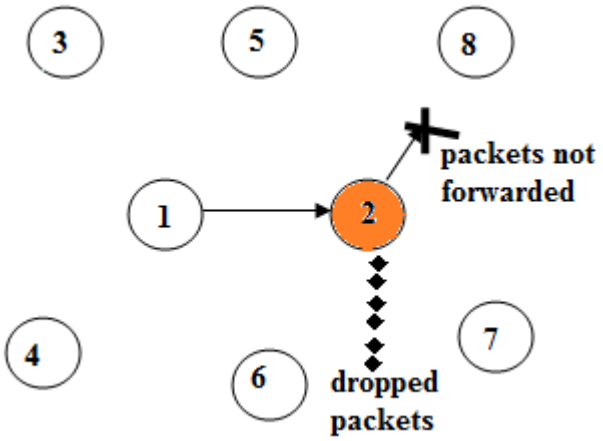


Figure 6: Selfish Node Behaviour

The selfish behaviour of a node is considered when it (S. Gupta, Nagpal, & Singla, 2011) (Akhtar & Sahoo):

- Simply drops the packets
- Does not co-operate in route establishment
- Blocks all the types of packet/traffic

- Refuses to forward the packet
- Advertises to the sender as the shortest route to the destination node

1.2. Detection Techniques for Selfish Node

Different types of techniques are used for detecting the misbehaving nodes (Liu H., Delgado-Frias José G., & S., 2007) (Kumar & Saraswathi, 2012):

1.2.1. Credit Based Approach

It discourages a node to attain selfish attitude by giving virtual money or credits called nuggets when a node forward packets of others because to send or receive its own packets the node requires enough credit. A nugget counter is incremented monotonically whenever it forwards the data packet for others (Sun, Chen, & Ku, 2012). When a node wants to send its own packet, it requires enough credit because if it is less than certain threshold it is not allowed to send packets.

- Buttyan and Hubaux used this concept of credits i.e. nuggets (beans) as payments for packet forwarding using two models: **Packet Purse Model** and **Packet Trade Model** (Liu H. *et al.*, 2007) (Ukey & Chawla, 2010) .
- Zhong et al. proposed a model **SPRITE** in which nodes keep receipt of the received and forwarded packets. There is centrally located (CSS) credit clearance system which decides the charge and credit for the reporting nodes (Liu H. *et al.*, 2007) (Usha & Radha, 2011).

The only drawback of this method is that it requires tamper proof hardware to maintain the nugget.

1.2.2. Reputation Based Approach

In reputation-based method, the network nodes collectively detect and declare the misbehaviour of suspicious nodes. Hence, the reputation or rating of the node is defined by the participation seen by other nodes i.e. according to the behavioural pattern. (Usha & Radha, 2011) (Liu H. *et al.*, 2007) (Liu, 2006).

- Marti Giuli and Baker proposed a reputation based mechanism having two modules- **watchdog** and **pathrater** added on every single node. The first module WATCHDOG maintains a buffer of recently sent/forwarded packets. This buffer is cleared only when the watchdog has overheard that the same packet has been forwarded by the next hop node over the medium and if the data packet remains in the buffer for a long time, the next hop neighbour is

suspected to be misbehaving. Based on watchdog's suspicion, PATHRATER module maintains a rating for every other node in the network and calculates the path metric by taking average of the node ratings in the path and then selects the best path (Ukey & Chawla, 2010) (Tamilarasan, 2011).

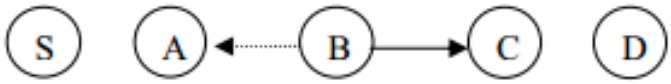


Figure 7: Watch Dog Mechanism

This mechanism fails to detect a misbehaving node in the presence of the ambiguous collisions, limited transmission power, receiver collisions, false misbehaviour conclusion and partial dropping (Liu H. *et al.*, 2007).

- Buchegger and Le Boudec proposed **CONFIDANT** protocol- based on selective altruism and utilitarianism (Padiya, Pandit, & Patel, 2013). The trust relationships and routing decisions are based on experienced, observed, or reported routing and forwarding behaviour of other nodes (Ukey & Chawla, 2010). It consists of four modules: Monitor, Reputation System, Path Manager, and Trust Manager (Tamilarasan, 2011). Their task is divided into two sections: process to handle their own observations and process to handle reports from trusted nodes.
- Michiardi et. al proposed the **CORE** technique for selfish node detection. CORE protocol permits for the positive reports to pass through but CONFIDANT protocol allows for the negative ones. Hence CORE prevents false reports as well as DOS attack. When a node doesnot cooperate, it is being given a negative rating, hence its reputation decreases (Tamilarasan, 2011) (Padiya *et al.*, 2013). CORE consists of Reputation table having data structure stored in each node and Watchdog mechanism to detect misbehavior nodes. Each row of the table comprises of unique identifier of the entity, collection of recent subjective observations made on that entity's behavior, list of the recent indirect reputation values provided by other entities and value of the reputation evaluated for a predefined function.
- Bansal et. al. proposed a stand-alone architecture **OCEAN** protocol which relies on its observation for avoiding the new vulnerability of the false accusation from the second-hand reputation exchanges. It classifies routing

misbehavior into the following two categories: misleading and selfish. If the node participates in the route discovery phenomenon but does not forward the data packet, then it is considered as misleading node as it misleads other nodes to route packets through it. If the node does not even take part in the route discovery process, it is referred as the selfish node (Tamilarasan, 2011).

- **SORI** based on detect and exclude mechanism comprises of following three components- Neighbor monitoring, Reputation propagation and Punishment (Tamilarasan, 2011). Each neighbor forwarding function is linked with two parameters - Request for forwarding (RF_n) and Has forwarded ($HF_n(x)$). It makes use of two records - Local evaluation record (firsthand trust) created using values of RFn and HFn(x) for depicting the confidence metric and Overall evaluation record is calculated when each node 'n' periodically updates its local evaluation record (S. Gupta, Nagpal, & Singla, 2011).

1.2.3. Acknowledgement Based Approach

The term "Acknowledgement" refers to informing the sender of the data packet about the successful arrival of the packet by destination. Varied approaches have been given related to acknowledgement methods, such as:

- In the Transmission Control Protocol, the **end-to-end acknowledgment** scheme (NACK) is employed. Such acknowledgments are sent by the receiver to inform the sender about the successful reception of the data packets up to certain locations of the continuous datastream. It increases routing overhead and end- to-end delay as acknowledgement packet has to be sent for each data packet received (Sun *et al.*, 2012).
- **Selective Acknowledgment** (SACK) technique is used for acknowledging out-of-order data blocks i.e. for certain data packets, acknowledgement packets are being sent from end receiver to the source node. Thereby, reducing the routing overhead to certain extent in comparison to end-to-end acknowledgement scheme. The limitation of SACK scheme is that acknowledgement packet has to travel from the end receiver towards the source, thereby increasing end-to-end delay for the data packet received (Ukey & Chawla, 2010).

- **TWO-ACK** scheme works on the basis of two-hop route, i.e. a set of triplets (N_1, N_2, N_3) is being considered between the source and destination (K. Balakrishnan, J. Deng, & P. K. Varshney, 2005) (Sun *et al.*, 2012) (Liu, 2006). The TWO-ACK packet is sent in opposite direction to the direction of data traffic, i.e. from the N_3 towards N_2 which is being forwarded to N_1 notifying N_1 that N_3 has received the packet successfully. The main drawback of this scheme is that the acknowledgement packet has to be sent for every data packet received over the two-hop route.

Figure 8: TWO-ACK Scheme

- **S-TWOACK** scheme minimizes the overhead caused by TWO-ACK approach as each TWOACK packet in the S-TWOACK scheme acknowledges the receipt of a number of data packets (K. Balakrishnan, J. Deng, & P. K. Varshney, 2005). But the limitation of this scheme is that it has no authentication mechanism for the acknowledgement packets to be sent from the receiver to the source node.
- **2ACK** scheme, a modified version of TWO-ACK scheme, reduces the overhead by sending 2ACK packets only for a fraction of data packets received (Sun *et al.*, 2012) (Liu, 2006). It tries to reduce the overhead by sending genuine and authenticated 2ACK packets only for a fraction of data packets received (Sun *et al.*, 2012) (Gupta & Chopde, 2013) (Liu, 2006). But it adds overhead and increases the end-to-end delay due to the authentication process carried out.

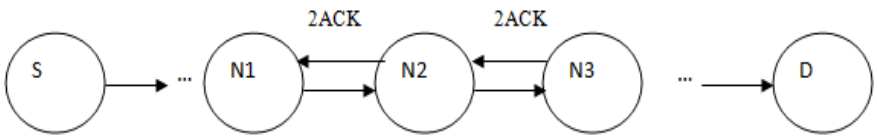


Figure 9: 2ACK Scheme

Hence, the 2ACK scheme is different from TWOACK and S-TWOACK scheme w.r.t. to the following reasons -

- The receiving node in the 2ACK scheme sends the 2ACK packets for a fraction of received data packets only while in the TWOACK scheme, TWOACK packets are being sent for every data packet received. Hence, sending the acknowledgement of the fraction of the data packets that have received gives 2ACK scheme a better performance w.r.t routing overhead.
- 2ACK scheme has an authentication mechanism to maintain genuineness of 2ACK packets.

Working of 2ACK scheme

Suppose that there are three consecutive nodes (triplet) N_1, N_2, and N_3 along a route. The route from the source node 'S' to the destination node 'D' is being generated in the route discovery mechanism of the DSR protocol. When the node N_1 sends a data packet to the node N_2 and the node N_2 forwards the data packet to the node N_3, then N_1 is not aware whether the node N_3 has received the data packet successfully or not. Such an ambiguity prevails even when there are no misbehaving nodes in the network. The problem becomes much more complex and severe in the open MANETs with such potential misbehaving nodes.

The 2ACK scheme makes use of an explicit acknowledgment to be sent by the node N_3 to inform the node N_1 of the successful reception of the data packet. When the data packet has been successfully received at node N_3, it then sends a 2ACK packet over two hops to the node N_1 with the ID of the corresponding data packet in the opposite direction of the routing path as shown in the figure 9 (Liu H. *et al.*, 2007) (Gupta & Chopde, 2013).

The triplet set [N_1, N_2, N_3] has been derived from the route of the original data traffic. Such a triplet set is used by the node N_1 to analyse and monitor the link $N_2 \rightarrow N_3$. The node N_1 in the triplet acts as the receiver of the 2ACK packet or the observing node and the node N_3 acts as the sender of the 2ACK packet. Such transmission of the 2ACK packet occurs for every set of triplets along the route. Therefore, the first node from the source will not act as a 2ACK packet sender and the last node just before the destination and the destination itself will not act as 2ACK receivers. To detect such misbehaviour, the 2ACK packet sender will

maintain a list of IDs of the data packets that have been sent but till now they have not been acknowledged. After the node N_1 has sent the data packet on the specified path, it then adds the data ID to LIST - the data structure which is maintained by the observing node or the receiver of the 2ACK packet. A counter of the forwarded data packets 'C_{pkts}' is being incremented simultaneously (Liu H. *et al.*, 2007).

At the node N_1, each ID of the data packet will remain on the list for λ seconds, the timeout for the 2ACK reception. If the 2ACK packet which corresponds to this ID reaches before the expiry of the timer, then this ID will be removed from that list. Otherwise, at the end of the timeout interval the the data ID will be removed and the counter referred as C_{mis} will be incremented. When the node N_3 receives the data packet, it will first determine whether there is a need to send the 2ACK packet to the node N_1. For reducing the additional routing overhead, only a certain fraction of the data packets are being acknowledged via 2ACK packets. This fraction is referred as the acknowledgment ratio, R_{ack} (Liu H. *et al.*, 2007). The node N_1 will observe the behaviour of the link $N_2 \rightarrow N_3$ for a certain time period say T_{obs} seconds. When the observation period finishes, the node N_1 will calculate the value for the C_{mis}/C_{pkts} as the ratio of missing 2ACK packets and then compares it with 'R_{mis}' which is the threshold for determining the allowable ratio of the total number of missed 2ACK packets to the total number of sent data packets. If this ratio turns out to be greater than R_{mis}, then the link $N_2 \rightarrow N_3$ is declared as the misbehaving link. The node N_1 will then send a misbehaviour report or the RERR packet. As the certain fraction of the received data packets are being acknowledged, R_{mis} must satisfy the following equation (Liu H. *et al.*, 2007) (Gupta & Chopde, 2013):

$$R_{mis} > 1 - R_{ack}$$

Each node which is receiving or overhearing such an RERR message will mark the link $N_2 \rightarrow N_3$ as the misbehaving link and then add it to the maintained blacklist of such misbehaving links. Later, when a node begins with its own data traffic, it will avoid using such misbehaving links which have become a part of its route.

This scheme overcomes the drawbacks of watch dog mechanism, i.e. the problem of ambiguous collisions, receiver collisions, limited transmission power and limited overhearing range. The major drawback of this scheme is that the

routing overhead is affected due to the number of authenticated 2ACK packets. Increase in the ratio of 2ACK packets and data transmissions results in increase of routing overhead. Also, 2ACK scheme is less resistant to the collusion attacks and malicious alarms (Liu H. *et al.*, 2007) (Gupta & Chopde, 2013).

Authentication technique

To authenticate the 2ACK packets, message authentication code function (Liu H. *et al.*, 2007) (Gupta & Chopde, 2013) is being used. MAC is a function of message and secret key producing a fixed length value. MAC algorithm even referred as keyed cryptographic hash function accepts as input a secret key and an arbitrary-length message to be authenticated, and outputs the MAC (even referred as a tag). This technique assumes two communicating parties (say A and B) sharing a secret key. When A has to send message to B, it calculates the MAC function (Stallings, 2007) -

$$MAC = C(K, M)$$

where M is the input message, C is the MAC function, K is the shared secret key, and MAC is the message authentication code. The message plus MAC are being transmitted to the intended recipient. The recipient then performs the same calculation on the message received using the same secret key to generate a new MAC. The received MAC is then compared with the calculated MAC. The MAC value protects both a message's data integrity as well as its authenticity, by allowing the verifiers (possessing the secret key) to detect the changes in the message content (Liu, 2006).

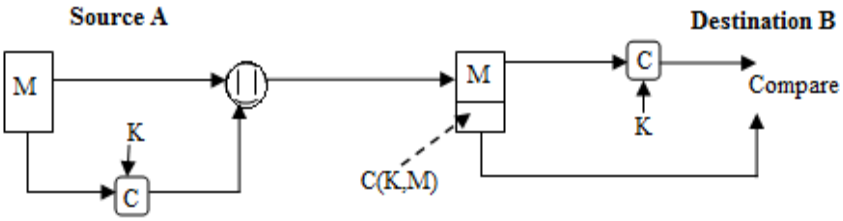


Figure 10: Mechanism of Message Authentication Code

CHAPTER 2
LITERATURE REVIEW

The routing protocols for ad-hoc networks based on a certain set of parameters has been studied and evaluated giving an overview of the characteristics, functionalities, comparison, merits and demerits of these protocols **(Ismail & Jaafar, 2007)**. The authors have not clearly mentioned that which particular algorithm or class of algorithm is best suited for all scenarios.

The fundamentals of ad hoc network i.e. the concept, features, status and vulnerabilities of MANET have been emphasized in the work **(Goyal, Parmar, & Rishi, 2011).** They even threw certain light on the routing protocols, challenging issues, emerging applications and future trends of MANET.

The susceptibility of MANET to different kinds of attacks on network and lower layers and their countermeasures have been presented **(Jain & Tokekar, 2011)**.

The research work **(Jhaveri, Patel, & Jinwala, 2012)** throws light on the unique characteristics of MANET that make it vulnerable to different types of attacks in different layers of protocol stack. They discussed about the serious threats like wormhole attack, black hole attack, grayhole attack etc. along with the solutions for detection and prevention of these attacks.

The state of art of the security issues in MANET have been investigated. In particular, the authors examined the routing attacks like link spoofing, colluding misrelay attacks and their countermeasures along with their advantages and disadvantages. They laid emphasis on the trade-off between the cost of implementing the particular solution and the effectiveness of security **(Kannhavong, Nakayama, Nemoto, Kato, & Jamalipour, 2007)**.

The comparison of both the reactive protocols- DSR and AODV has been done to determine which protocols works better than the other. It resulted in a conclusion that the packet delivery ratio is almost same in both the protocols but the effect of routing overhead makes DSR a preferable choice than AODV. Hence, the effect of non co-operation or misbehaviour of nodes also affects the percentage of routing overhead in these protocols **(Moses, 2012)**.

Significant light has been thrown on the nature and characteristic of the selfish node as well as the overall impact of selfish node concentration in MANETs. When the concentration of selfish nodes increases, the network parameters become poorer and poorer but the network never comes to halt **(S. Gupta, Nagpal, & Singla, 2011)**.

The authors have presented an overview of most of the innovated techniques to detect selfish nodes which do not consume the energy resources like battery, CPU power and bandwidth for the retransmission of the data of the other nodes but reserve these resources only for themselves **(Padiya, Pandit, & Patel, 2013)**.

A logical survey for detection of the misbehaving nodes in mobile ad hoc network using Intrusion Detection System depicts that the encryption and authentication solution being considered as the first line of defence are not enough sufficient to protect MANETs **(Tamilarasan, 2011)**. Therefore, Intrusion Detection Systems are needed to be the second line of defence for protecting the network security.

After studying about the impact of misbehaving nodes on the efficiency, reliability and fairness in MANET, the authors **(Wu & Yu, 2010)** proposed a threshold based method to increase selfish node detection rate and decrease false detection rate.

Selfishness test was conducted on the selfish nodes resulting in the presentation of the mathematical model for the selfish node detection using the probability density function **(Akhtar & Sahoo)**. This model formulates the selfishness problem with the use of prior probability as well as continuous Bayes' theorem giving accurate results as it uses heuristic model rather than deterministic.

A game theoretic routing model 'STACRP' **(Chatterjee, Sengupta, & Ghosh, 2012)** based on both the credit and reputation based approach was proposed for trusted framework in MANET. It is a lightweight model in terms of computational cost and communication requirements to mitigate selfishness and enforce co-operation between the nodes.

A new protocol LOTTO **(Wang & Singhal, 2007)** for route discovery in MANET with selfish nodes was proposed showing that it provides high packet delivery ratio with low overhead and lower end-to-end delay in comparison to the ad-hoc VCG.

In the work, the authors stated that to simulate co-operation between nodes in MANET, reimbursement of the forwarding nodes according to their cost (incentive)

is done. But, selfish node may still cheat to maximize their payoff. So the authors gave a new protocol TMRP based on extended version of AOMDV. This protocol tries to achieve load balancing in addition to high throughput and low overhead without compromising truthfulness **(Wang, Giruka, & Singhal, 2008)**.

The impact of selfish node from the perspective of replica allocation has been brought to the limelight resulting in the reduction of overall data accessibility. A selfish node detection algorithm considering partial selfishness and novel replica allocation techniques **(Choi, Shim, Lee, & Wu, 2012)** has been proposed to properly cope with selfish replica allocation.

Inspired by the work done in (Choi *et al.*, 2012), the researchers proposed a combined credit risk & collaborative watchdog method for detecting the selfish node and applied the SCF tree based replica allocation method for handling the selfish replica allocation **(Kumar & Saraswathi, 2012)**.

The work **(Rebahi, Mujica-V, & Sisalem, 2005)** provides a trust model extending routing protocol based on the reputation concept. This model is based on the reputation module and trust module. The main drawback of this method is that it works only for detecting the packet dropping misbehaviour without discarding the malicious nodes.

Some researchers have extended the work upon watchdog mechanism for selfish node detection using a collaborative approach. They have proposed an analytical model for detecting time and cost which makes use of Continuous Time Markov Chain **(Hernandez-Orallo, Serrat, Cano, Calafate, & Manzoni, 2012)**. This model fails to evaluate the effect of false positives and false negatives.

A hardware based cache scheme **(Lin, Delgado-Frias, & Medidi, 2007)** has been presented to detect selfish nodes in mobile ad hoc network. The hardware does monitoring of the activities of the upper layer software and reports about the software misbehaviour to the other mobile nodes in the network. The identity information of recently received packets is stores in the hardware cache. This cache is used by the detection mechanism to tell the difference between original packet and duplicate packet which seems to have low detection effectiveness in the case of selective dropping scenario.

Another hardware based two-timer scheme to detect the misbehaving nodes was proposed. Detection Timer and Reward Timer are two timers kept in to consideration for detection of the misbehaving nodes and gives reward to the well behaving nodes respectively **(Lui, Delgado-Frias, & Medidi, 2007)**. This mechanism results in high detection of misbehaving nodes and low false positives.

The work **(Ukey & Chawla, 2010)** deals with the various acknowledgement based schemes. The authors proposed a new technique in which nodes in the network are logically grouped and then within a group, the acknowledgement is sent from the receiver to the source. Then the acknowledgement is sent from one group to the other group.

Mitigating the selfish node behaviour by NACK scheme which includes two techniques- acknowledgement based and timestamp comparison has been worked upon **(Sun, Chen, & Ku, 2012)**. The NACK scheme detects collusion attacks in addition to prevention of routing misbehaviour.

An end-to-end ACK scheme and Collective Network Arbitration Protocol was provided to counter the misbehaviour patterns in a MANET without compromising the throughput of the network. The nodes tend to be less enthusiastic over forwarding packets of neighbouring nodes to their destination. Hence, they tamper with their back off values generating values that are more non-random in nature. A back off interval is generated for which a node is expected to wait **(Usha & Radha, 2011)**. A new algorithm was proposed with an aim to minimize false detection of sender or receiver nodes.

Two network layer acknowledgement based add-on schemes termed TWO-ACK and S-TWOACK have been proposed **(Balakrishnan et al., 2005)**. The comparative analysis has been done for both the schemes resulting in the increased expectancy of false alarm rate without any increase in the routing overheads in S-TWOACK scheme.

The routing misbehaviour due to the presence of selfish nodes has been studied. These nodes participate in the route discovery and route maintenance mechanism but they refuse to forward data packets to other nodes. Therefore, the 2ACK scheme was proposed for detecting routing misbehaviour and mitigating their

adverse effect **(Liu, 2006)**. So to reduce the additional routing overhead, 2ACK scheme acknowledges only a fraction of the received data packet.

The problem of misbehaviour on black hole attack has been focussed in the work **(Liu, Deng, Varshney, & Balakrishnan, 2007)**. A new algorithm based on the acknowledgement approach- 2ACK scheme was implemented on DSR protocol. This in turn works for providing authentication to the acknowledgement packets and overcoming the problems of receiver collisions, ambiguous collisions and transmission power.

The comparative analysis of the 2ACK scheme implemented over DSR protocol has been done. The results show that the authentication of 2ACK packets with one way hash chaining technique outperforms with respect to 2ACK packets authenticated by Digital Signature approach **(Tamilarasi & Sundararajan, 2012)**.

In the work **(Chobe & Gothawal, 2013)** 2ACK scheme was implemented over AOMDV protocol to remove the selfishness that existed earlier in AODV protocol. They did comparison for 2ACK on AODV and 2ACK on AOMDV for misbehaviour detection and evaluated values for parameters- packet delivery ratio, delay and routing overhead.

An add-on technique has been proposed **(Vijaya, 2008)** to reduce the routing overhead in 2ACK scheme by acknowledging only a fraction of data packets in DSR protocol. The proposed method deals in multicasting in which the sender can broadcast to other nodes about misbehaving nodes. Therefore, other nodes can avoid that path and take another path for data transmission.

The 2-ACK scheme for detecting and eliminating the selfish nodes by choosing the other path for transmitting the data was proposed. After choosing the other path for transmitting the data, large amount of routing overhead is generated. The authors have used a Novel Routing algorithm based on mobile software agents modelled on ants that collect and disseminate information about the location of nodes in the network. This is a key aspect of the GPSAL algorithm that helps to accelerate route discovery **(Gaikwad & Adane, 2013)**. This methodology decreases the routing overhead, makes the network stable and checks the confidentiality of data message.

A random 2ACK scheme has been given in the work **(Boopathi, Insozhan, & Vinod, 2013)** in which a fraction of data packets will be acknowledged to reduce the routing overhead problem. Still the development part is in progress taking DSR as the basic routing protocol.

The limitation of 2ACK scheme i.e. increased routing overhead due to authenticated 2ACK packets has been focussed **(Rathod, Ingle, Kankate, & Kawale, 2013)**. The researchers are still working upon reducing the routing overhead in DSR protocol by minimizing the acknowledgment ratio.

The work **(P. Gupta & Chopde, 2013)** laid emphasis on the performance degradation of the mobile ad hoc network due to selfish nodes. They have proposed improved 2ACK scheme over DSR for detecting the misbehaving link and reducing additional overhead by minimizing acknowledgment of the received data packets. The results of improved 2ACK scheme prove far better than the 2ACK scheme and NACK.

The authors studied about routing misbehaviour in OLSR protocol. The 2ACK scheme was applied on OLSR protocol for getting higher throughput and packet delivery ratio **(Wankhade, 2012)**. The problem of higher routing overhead was determined due to 2ACK packets.

CHAPTER 3

SIMULATION WORK

3.1. Simulator Software

NS2 is a discrete event simulator used for simulation of TCP, routing and multicast protocols over all networks (Meeneghan & Delaney, 2004). It consists of two key languages: C++ and OTcl (Tcl script programming with object oriented extension). C++ serves as the backend i.e. it defines the internal mechanism of the simulation objects while the OTcl helps in setting up the simulation by the assembling and configuring of the objects as well as the scheduling of the discrete events. TclCL helps in linking of the C++ and the OTcl.

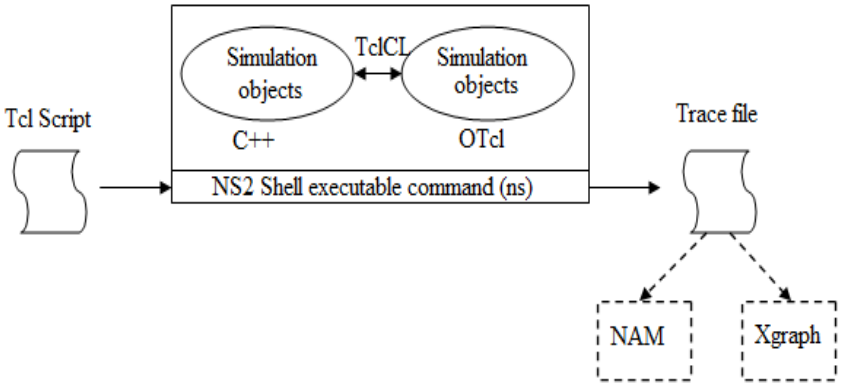


Figure 11: Basic Architecture of NS2

Basic NS2 programming contains (Issariyakul & Hossain, 2011) :

- Create the event scheduler
- Turn on tracing “tr file”
- Creating network i.e. routing, transport connection-agents, traffic etc.
- Monitoring which visualizes the network using “nam” i.e. network animator.

NS2 has following characteristics (Meeneghan & Delaney, 2004).:

- Router queue Management Techniques
- Multicasting
- Simulation of wireless networks
  - Terrestrial (cellular, adhoc, GPRS, WLAN, BLUETOOTH), satellite
  - IEEE 802.11 can be simulated, Mobile-IP and adhoc protocols such as DSR, TORA, DSDV and AODV.
- Traffic Source Behaviour- www, CBR, VBR

- Transport Agents- UDP/TCP
- Routing
- Packet flow
- Network Topology
- Applications- Telnet, FTP, Ping
- Tracing Packets on all links/specific links

Tool Command Language (Tcl) Script

Tcl script is a dynamic programming language.

After login into terminal, we follow the following steps—

1) vi filename.tcl
   It will open page to write tcl scripts.
2) Press esc-> colon (shift + semicolon) -> wq (save and quit) to save file.
3) Type “ns <filename>.tcl” in terminal to run tcl script.

Network Animator (Nam)

Nam is a Tcl/TK based animation tool used for viewing the network simulation traces and the packet traces (Meeneghan & Delaney, 2004). Nam has following characteristics:

- It is meant for providing visual interpretation of the network topology created.
- Nam control includes play, stop, forward, pause, rewind, packet monitor facility and a display speed controller.
- It can be directly executed from a Tcl script.
- It also provides information regarding throughput and number packets on each link.
- Nam provides drag and drop interface for creating topologies.

To run the NAM executable, following command is used:

nam <tracefile>

where, tracefile is the name of the trace file produced i.e.file.tr.

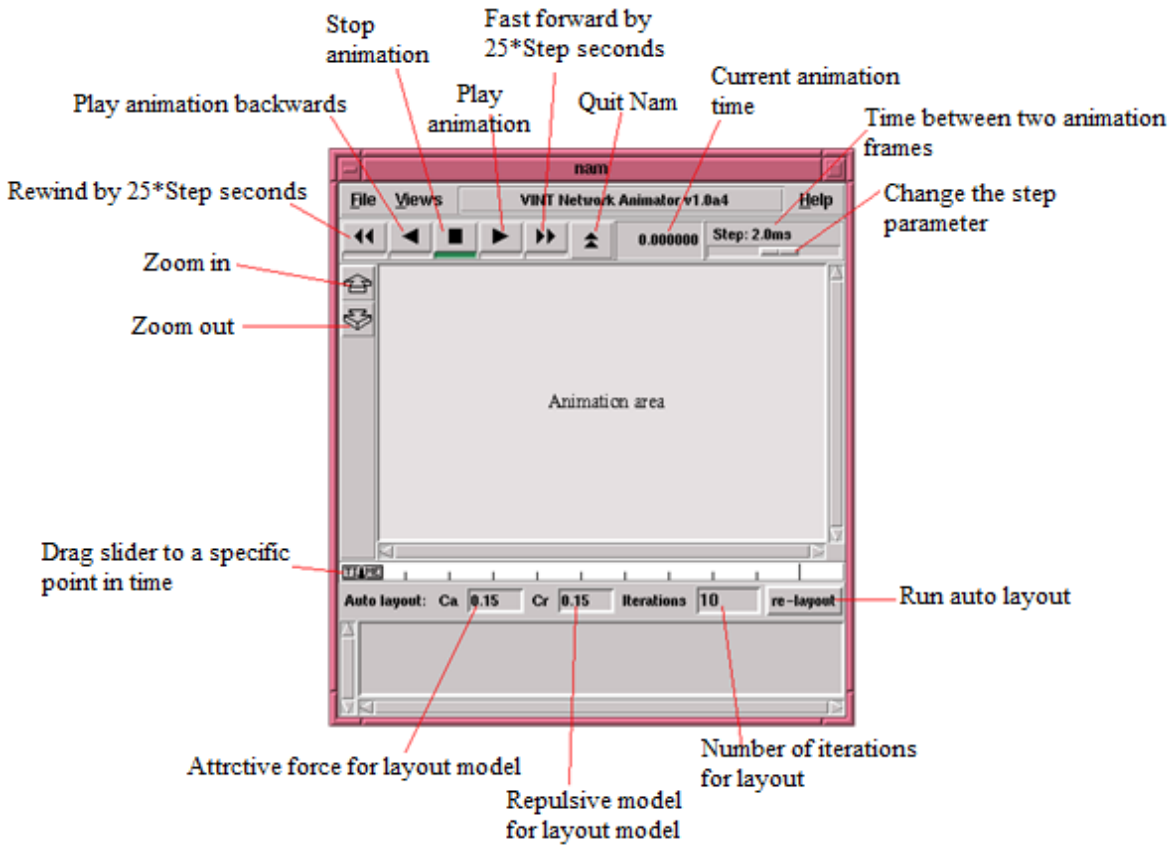


Figure 12: Nam Window

Trace File (tr)

Trace file contains all the information needed for the purpose of animation - both on a static network layout and dynamic events such as packet arrivals, departures, drops and link failures. A trace file data contains information like simulation time of event occurrence, packet type, packet size, IP flow identifier, source address, destination address, sequence number, unique packet identifier etc (Meeneghan & Delaney, 2004).

Type identifier	Time	Source node	Destinat-ion node	Packet name	Packet size	Flags	Flow id	Source address	Destinati-on address	Sequence number	Packet unique ID

Figure 13: Trace File Format

Awk Scripts

AWK Scripts (Meeneghan & Delaney, 2004) are used in processing the data from the log (trace files).

To execute awk script, the following command is used-

awk –f <awk-filename>.awk <tr-filename>.tr

Xgraph

Xgraph is X-Windows application that includes plotting and graphing. To use XGraph in NS-2, the executable can be called within a TCL Script with the help of following command- exec xgraph <filename>.tr -geometry <area-size>. But this loads a graph displaying visually the information of the trace file produced in the simulation (Meeneghan & Delaney, 2004). Hence, it is executed separately for less overhead.

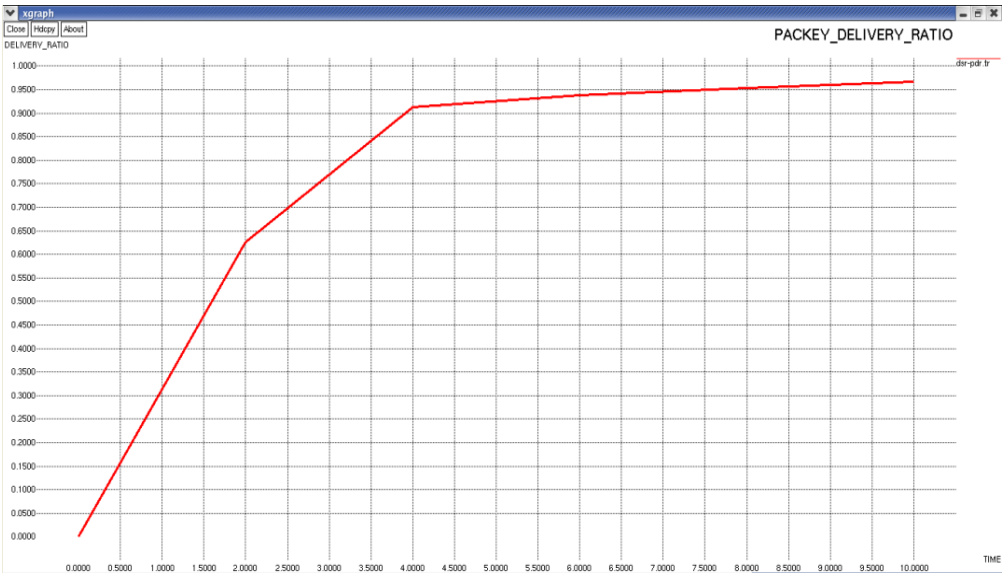


Figure 14: Xgraph in NS2

The main weakness of NS-2 is the lack of graphical representation of simulated output data. To process the data into suitable format for tools like “Xgraph”, the raw data must be processed using scripting language such as “awk script”. The NS-2 is not user friendly due to its text based interface.

3.2. Problem Statement

As co-operation between the mobile nodes is must for proper functioning of MANET, but sometimes nodes don’t co-operate with each other. They behave in a selfish manner. Hence, the overall performance and efficiency of the network degrades. The acknowledgement based scheme ‘2ACK’, when applied on the network for selfish node detection results in increased routing overhead due to the authenticated 2ACK packets.

3.3. Objective of the Work

This dissertation work aims to achieve the following objectives on the basis of the experiments to be performed -

- Selfish node attack will be analyzed to see its effect on the overall performance of the network
  - Variation in throughput due to the presence of selfish node
  - Number of packets lost by a selfish node
  - End-to-end delay caused
  - Routing overhead
- To implement existing 2ACK scheme for detection of misbehaviour.
- To implement an enhanced defence mechanism by modifying the authentication mechanism of existing 2ACK for increasing the network performance.

3.4. Methodology

The dissertation work has been carried out step by step as per the following phases:

Phase 1:

The basic topology of 50 mobile nodes in area size of 500*500 sq. m. having CBR traffic type, TCP connection agents, DSR protocol is created. The value of parameters- throughput, delay, packet delivery ratio, routing overhead and energy is evaluated for this scenario.

Phase 2:

Two selfish nodes dropping the data packets are introduced in the topology already set in phase 1. The value of the above mentioned parameters is calculated in this case also.

Phase 3:

2ACK scheme on DSR protocol is implemented for detecting the selfish nodes. The value of parameters is then analyzed for 2ACK scheme.

Phase 4:

Enhancement has been done to 2ACK scheme by applying RSA with “dynamic key generation” for reducing the end to end delay and routing overhead caused due to authenticated 2ACK packets. Dynamic key generation is the mechanism in

which for every new message, a new public key is being generated. So even if the intruder has come to know a single key used for the encryption/decryption of a particular message, he won't be able to encrypt/decrypt the other messages as the key is continuously changing for every new message. The value of the parameters are then calculated.

3.5. Parameters

The following parameter values are measured in this dissertation work:

a) Throughput

Throughput is defined as the total amount of data that reaches a receiver from a sender to the time it takes for the receiver to get the last packet. It is measured in bits per second (bit/s or bps).

b) Delay

Delay is calculated on the basis of time gap between the sending of the data packet by the constant bit rate source and its receipt at the corresponding constant bit rate receiver. It is measured in seconds.

c) Packet delivery ratio

Packet delivery ratio is referred as the ratio of the number of packets generated at the source node to the number of packets that have been received by the destination node.

d) Routing overhead

It is referred to as the total number of routing related transmissions (RREQ, RREP, RERR and 2ACK) transmitted by the routing protocol.

e) Energy

Energy is defined as the total/average power consumed by the network. It is measured in Joules.

CHAPTER 4

RESULTS AND DISCUSSION

The performance of the network has been evaluated by means of simulation using NS-2.34. The simulation parameters are set as under:

Table 2: Simulation Parameters

Channel Type	Wireless
Radio propagation model	Two Ray Ground
Antenna type	OmniAntenna
Interface Queue Type(IFQ)	CMUPriQueue
Routing Protocol	DSR
Interface type	WirelessPhy
Topology area	500*500 sq. m
Number of mobile nodes	50
Traffic type	Constant Bit Rate

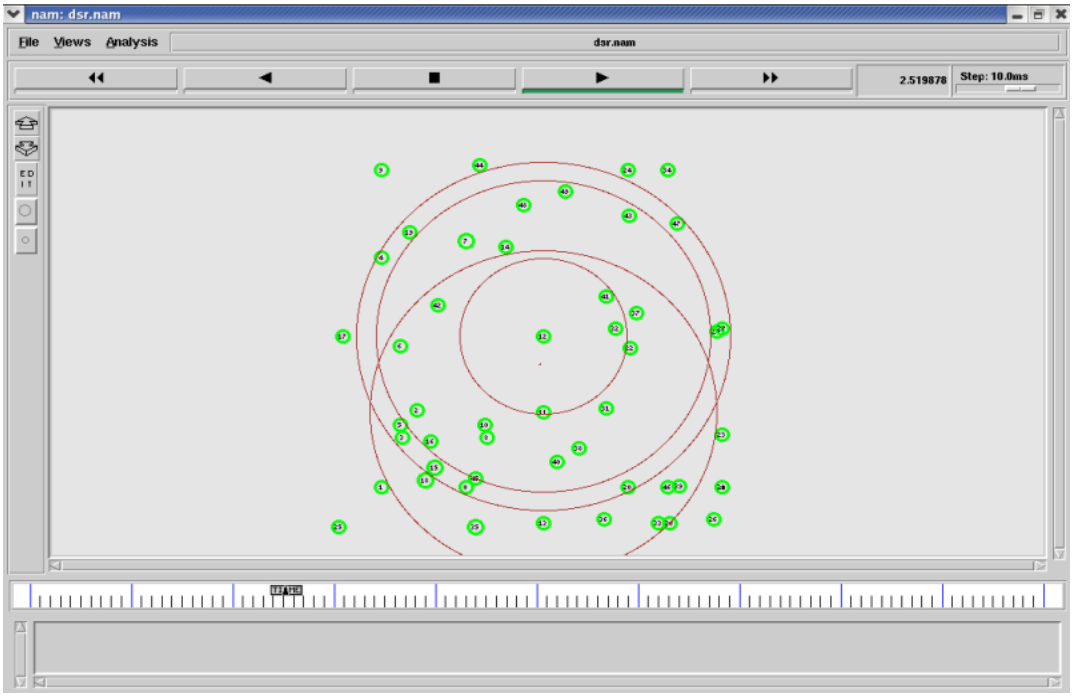


Figure 15: Simulation of the Topology of 50 Mobile Nodes

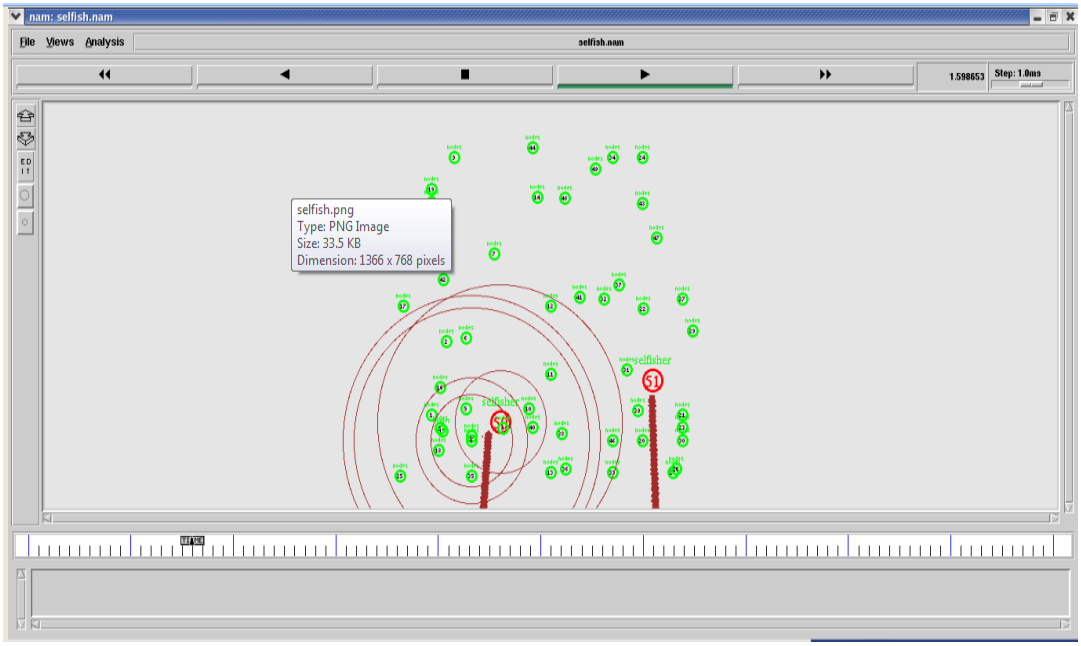


Figure 16: Simulation of the Topology with the Introduction of Selfish Nodes

Scenario 1:

The topology of 50 mobile nodes is set working in accordance with the DSR protocol having constant data packet transmission rate. The network is being analyzed for a particular time span.

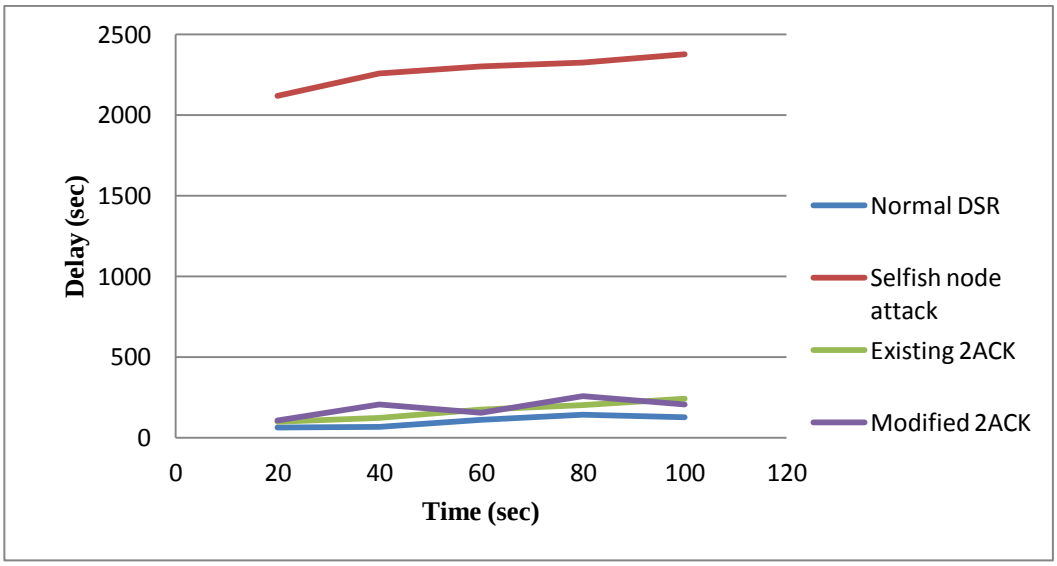


Figure 17: Analysis of Delay w.r.t. Time Period

Figure 17 depicts that in normal DSR phase, delay is showing a slight increase in throughout over the certain time period set for analysis. The introduction of selfish node in the network leads to large delay as the selfish nodes have the tendency for not participating in the packet forwarding process. When we apply existing

2ACK scheme for selfish node detection, the delay parameter is reduced to a great extent in comparison to the selfish node attack. The modified 2ACK scheme though successful in reducing routing overhead yet shows fluctuating values.

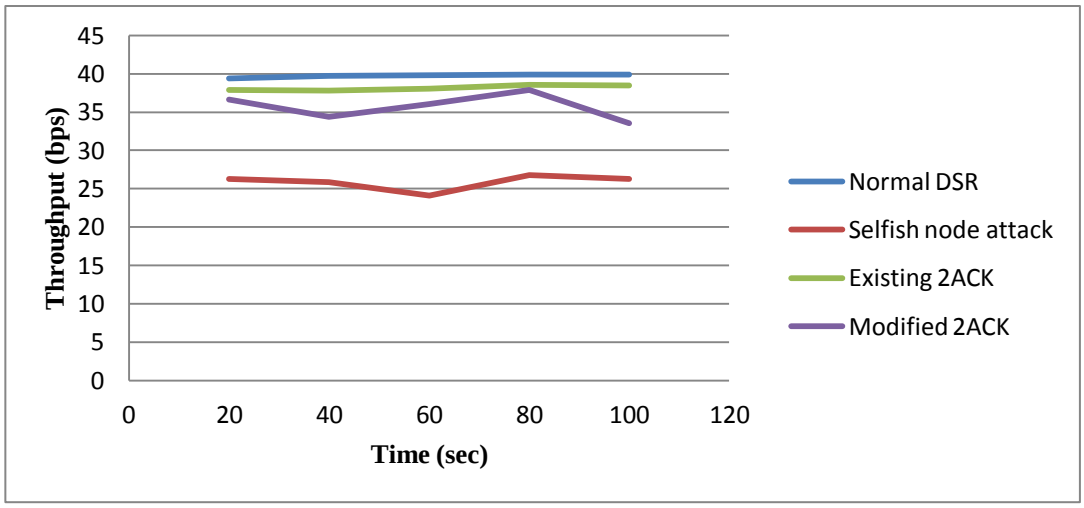


Figure 18: Analysis of Throughput w.r.t. Time Period

Figure 18 shows the comparative chart of throughput parameter for all the four phases we have worked upon. The throughput in normal DSR is slightly showing a gradual increase. When the network has selfish nodes in it, the throughput is reduced to a large extent as the selfish nodes drop most of the data packets. With the implementation of existing and modified 2ACK scheme, the throughput is improved comparatively to a great extent.

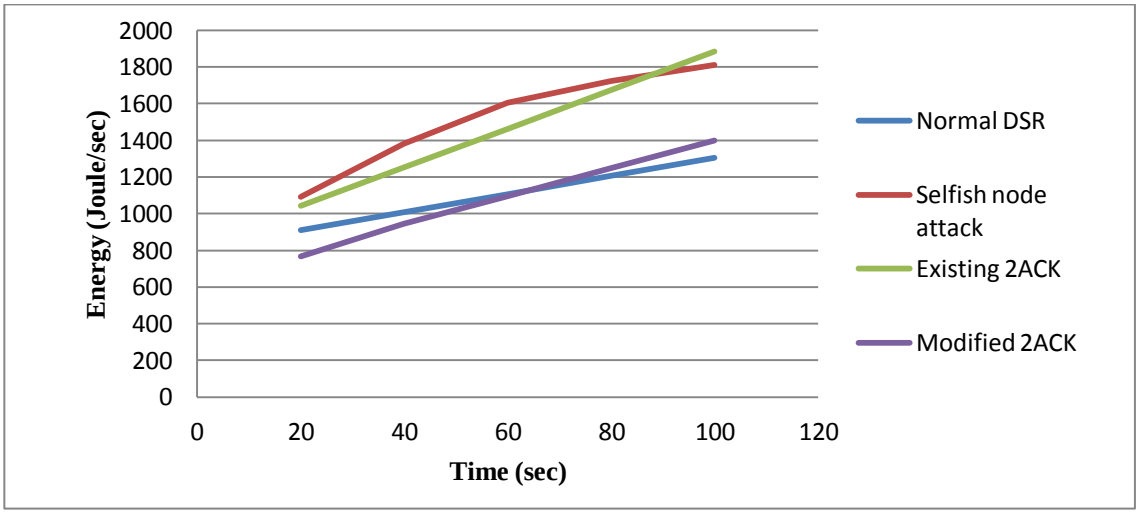


Figure 19: Analysis of Energy w.r.t. Time Period

The battery constrained nodes tend to lose their own energy level after certain interval of time. Thereby increasing the energy consumption. The selfish nodes

have the tendency to exhaust more energy resources of the network in order to preserve their own energy. The energy consumption is maximum in the selfish node attack. The existing 2ACK scheme tends to lower down the energy of the selfish nodes. With the modified 2ACK scheme, the goal of consuming minimum energy is achieved.

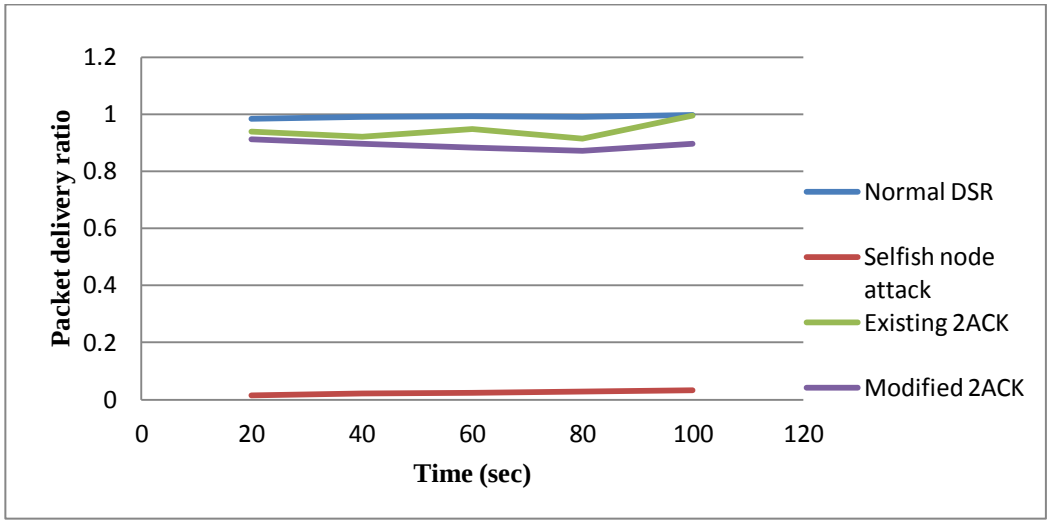


Figure 20: Analysis of Packet Delivery Ratio w.r.t. Time Period

From figure 20, we can infer that when the selfish node is present in the network, the packets are never delivered at the destined node. Hence the packet delivery ratio is less in comparison to the normal DSR. The existing 2ACK scheme and modified 2ACK scheme though have less throughput than the normal DSR scenario but still have been able to increase the throughput in comparison to the selfish node attack.

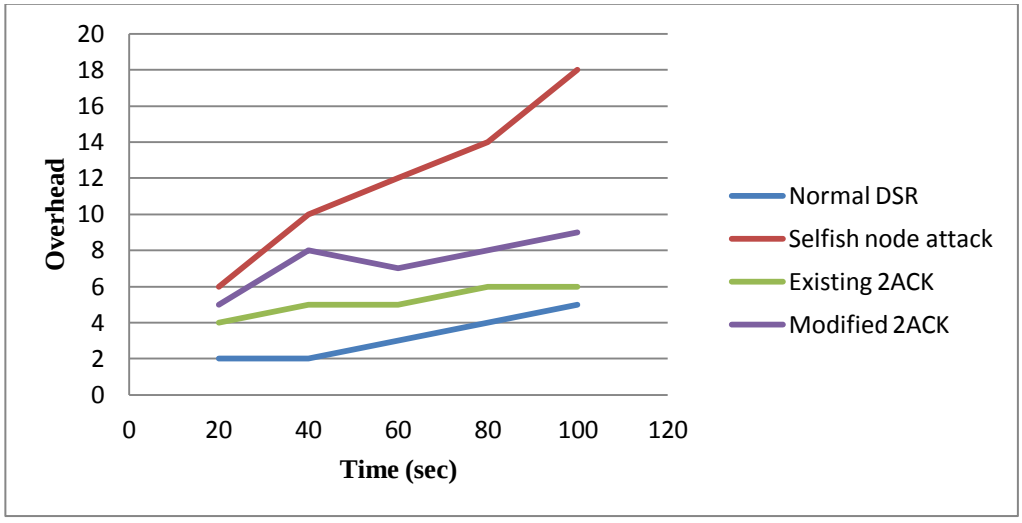


Figure 21: Analysis of Routing Overhead w.r.t. Time Period

The routing overhead depends on the number of routing or control related transmissions per node. When analyzed for a particular time span, we conclude that the overhead is increased manifold with the introduction of selfish node because these nodes send the false signals to the other nodes advertising themselves having the shortest path to the destination. Hence, the more number of RREQs, RREPs and even RERR cause congestion over the traffic than the normal DSR. In existing and modified 2ACK scheme, the 2ACK packets are also sent in addition to the control signals, so the routing overhead shows fluctuating value.

Scenario 2:

The topology set is analyzed with respect to varying packet size over the same simulation time.

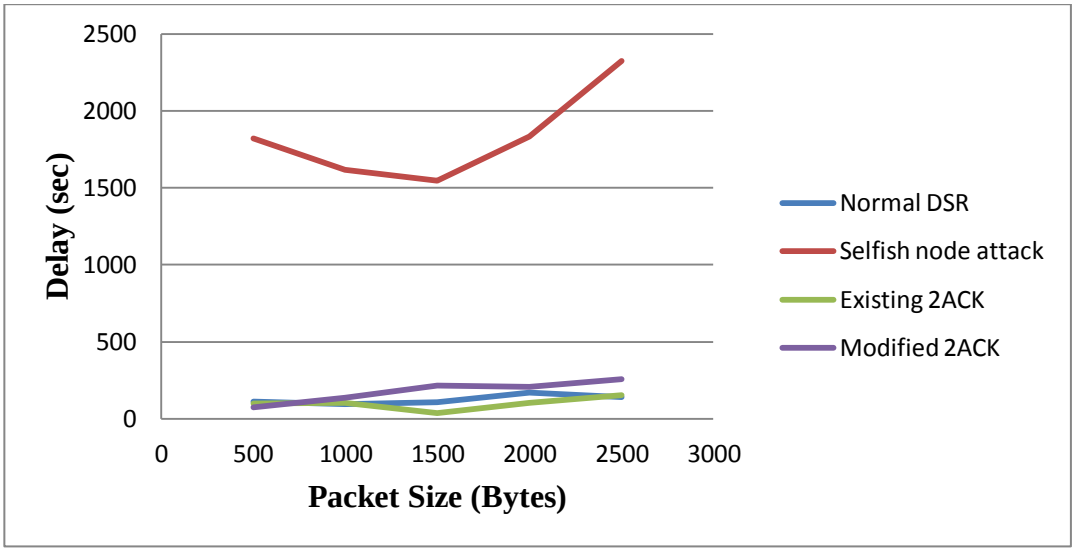


Figure 22: Analysis of Delay w.r.t. Packet Size

Figure 22 shows that with the increase in number of data packets, the delay generally goes on increasing. The selfish nodes may/may not forward the packets to the destination, so the delay value shows a uneven distribution. The value of delay is comparatively much higher in selfish node attack than the existing and modified 2ACK scheme.

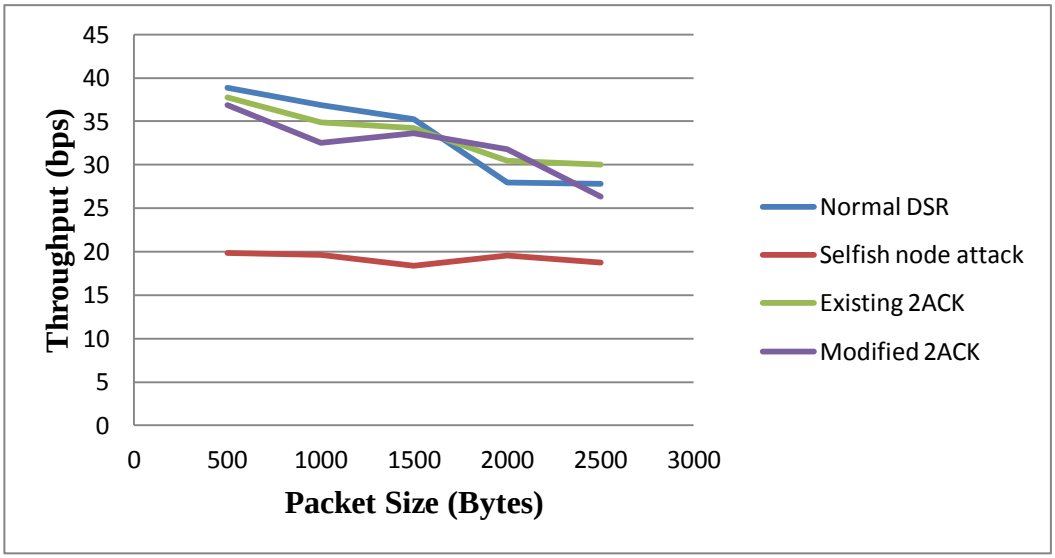


Figure 23: Analysis of Throughput w.r.t. Packet Size

For normal DSR phase, the throughput is gradually decreasing w.r.t packet size. The selfish nodes drop most of the packets, hence the throughput comes out to be low for varying packet size. The implementation of existing 2ACK scheme tries to enhance the throughput parameter in accordance with normal DSR phase. With the modified 2ACK scheme, the throughput shows fluctuating values.

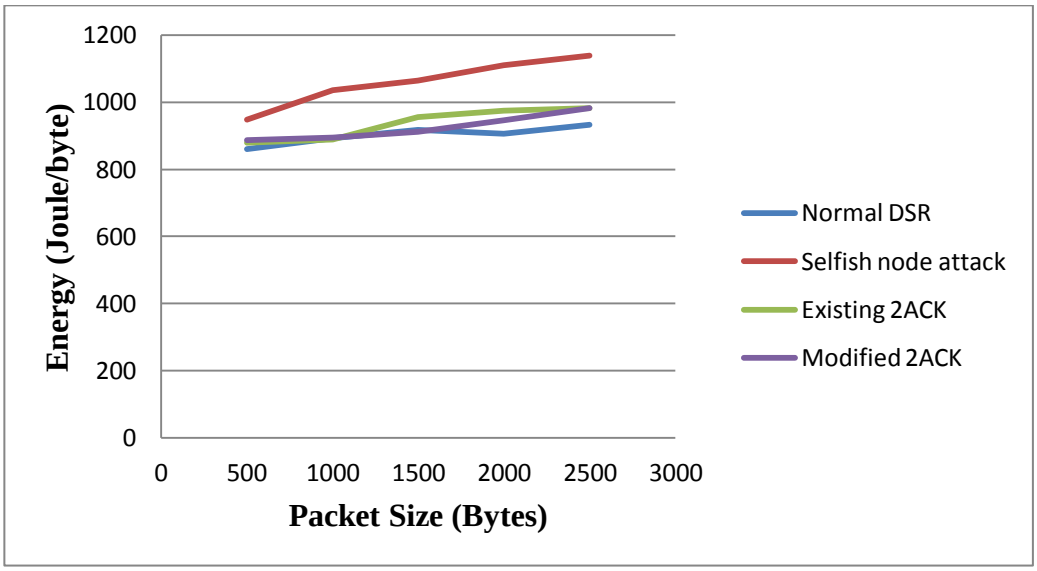


Figure 24: Analysis of Energy w.r.t Packet Size

Figure 24 depicts that with the increasing packet size, more and more energy is being consumed by the network. The normal DSR shows a gradual increase whereas the selfish node shows that large amount of energy is being consumed for varying packet size. This is due to the fact that these selfish nodes don't forward the packets rather they send fake route requests just to exhaust the other

nodes' energy resources. The existing 2ACK scheme has slightly more energy consumption than the modified 2ACK scheme which is far bit lower than the selfish node attack.

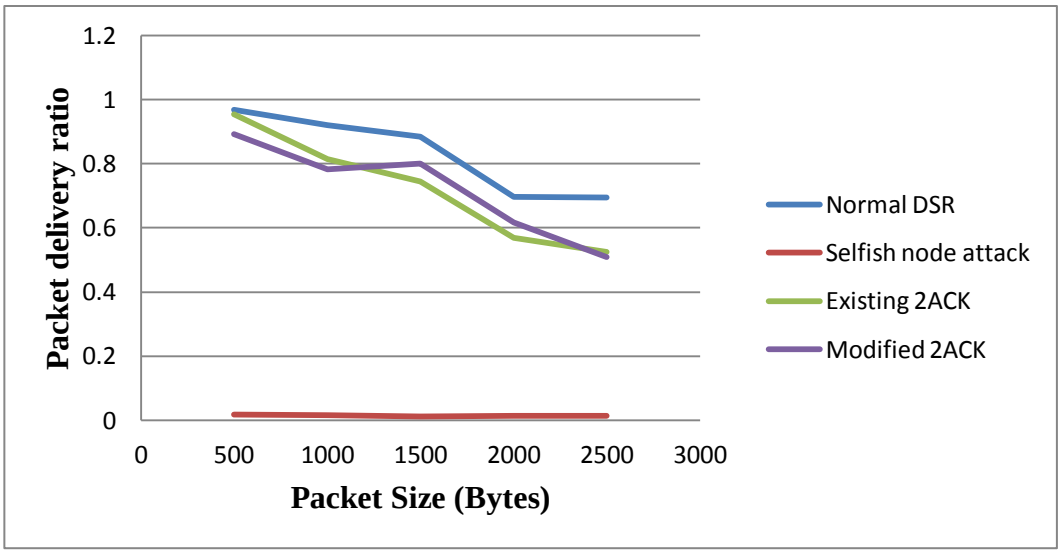


Figure 25: Analysis of Packet Delivery Ratio w.r.t Packet Size

The normal DSR shows the maximum packet delivery ratio while the selfish node attack has the minimum packet delivery ratio. The existing and modified 2ACK scheme show the decreasing level of packet delivery ratio when the number of packets keep on increasing.

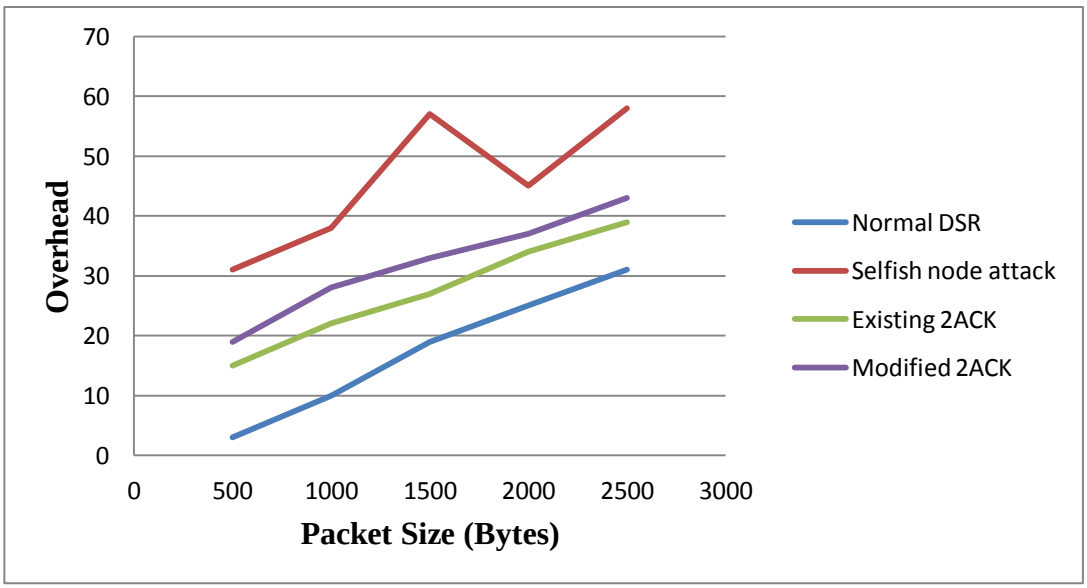


Figure 26: Analysis of Routing Overhead w.r.t Packet Size

From figure 26, we can infer that the routing overhead increases with the simultaneous increase in the packet size. In normal DSR, the routing overhead is

very low while in selfish node attack the routing overhead is very high because of the fake RREQs and RREPs sent by the selfish nodes. The existing 2ACK scheme involves sending of 2ACK packets in addition to other route messages, hence it has comparatively higher overhead than the normal DSR case. But the modified 2ACK scheme fails to reduce the routing overhead in relation to the existing one.

CHAPTER 5
CONCLUSION

Selfishness has become a major problem in the mobile networks. This dissertation work focuses on the selfish node attack and its overall effect on the network performance. The goal of implementation and analysis of the selfish behaviour has been achieved. The acknowledgement based scheme ‘2ACK’ has been used for detecting the selfish behaviour. Hence, resolving the problem of selfishness to a certain extent. The major limitation of increased routing overhead due to authenticated 2ACK packets has been worked upon. Instead of hash chaining mechanism used as MAC in 2ACK scheme, we have implemented RSA algorithm with dynamic key generation method for improving the performance of the network. The end-to-end delay is nearly same as the existing 2ACK scheme. The energy consumption has been lowered with the use of RSA with dynamic key generation. The attempt to reduce routing overhead could not turn out to be successful. In coming future we would be working on mechanisms for better network performance.

REFERENCES

Akhtar, M. A. K., & Sahoo, G. Mathematical Model for the Detection of Selfish Nodes in MANETs. *International Journal of Computer Science and Informatics,* 1(3), 25-28.

Balakrishnan, K., Deng, J., & Varshney, P. K. (2005). TWOACK: Preventing Selfishness in Mobile Adhoc Networks. Wireless Communications and Networking Conference, 2005 IEEE. Vol. 4, pp. 2137-2142. IEEE.

Basagni, S., Conti, M., Giordano, S., & Stojmenovic, I. (Eds.) (2004). *Mobile Adhoc Networking*. John Wiley & Sons.

Boopathi, G. M., Insozhan, N., & Vinod, S. (2013). Selfish Nodes Detection Using Random 2ACK in MANETs. *International Journal of Emerging Science and Engineering,* 1(4), 3-5.

Boukerche, A. (2008). *Algorithms and Protocols for Wireless, Mobile Adhoc Networks*. Vol. 77. John Wiley & Sons.

Chatterjee, P., Sengupta, I., & Ghosh, S. K. (2012). STACRP: A Secure Trusted Auction Oriented Clustering Based Routing Protocol for MANET. *Cluster Computing,* 15(3), 303-320.

Chobe, S. N., & Gothawal, D. (2013). An Acknowledgement Based Approach for Routing Misbehavior Detection in MANET With AOMDV. *International Journal of Advanced Computational Engineering and Networking,* 1(5), 5-10.

Choi, J.-H., Shim, K.-S., Lee, S., & Wu, K.-L. (2012). Handling Selfishness in Replica Allocation over a Mobile Adhoc Network. Mobile Computing, IEEE Transactions on, 11(2), 278-291.

Gaikwad, S., & Adane, D. (2013). Reduction in Routing Overhead in MANET Using 2-ACK Scheme and Novel Routing Algorithm. *International Journal of Engineering Trends and Technology,* 4(8), 3677-3681.

Goyal, P., Parmar, V., & Rishi, R. (2011). MANET: Vulnerabilities, Challenges, Attacks, Application. *International Journal of Computational Engineering & Management,* 11, 32-37.

Gupta, P., & Chopde, S. (2013). Detection of Routing Misbehavior in MANET Using Improved 2ACK. *IOSR Journal of Computer Engineering,* 9(1), 53-60.

Gupta, S., Nagpal, C., & Singla, C. (2011). Impact of Selfish Node Concentration in MANETs. *International Journal of Wireless & Mobile Networks,* 3(2), 29-37.

Hernandez-Orallo, E., Serrat, M. D., Cano, J.-C., Calafate, C. T., & Manzoni, P. (2012). Improving Selfish Node Detection in MANETs Using a Collaborative Watchdog. Communications Letters, IEEE, 16(5), 642-645.

Ilyas, M. (2002). *The Handbook of Adhoc Wireless Networks*. CRC Press.

Ismail, D., & Jaafar, M. (2007). Mobile Adhoc Network Overview. Applied Electromagnetics, 2007 Asia-Pacific Conference on, pp. 1-8, Melaka.

Issariyakul, T., & Hossain, E. (2011). *Introduction to Network Simulator NS2*: Springer.

Jain, A. K., & Tokekar, V. (2011). Classification of Denial of Service Attacks in Mobile Adhoc Networks. Computational Intelligence and Communication Networks (CICN), 2011 International Conference on, pp. 256-261, Gwalior, India.

Jhaveri, R. H., Patel, S. J., & Jinwala, D. C. (2012). DoS attacks in mobile ad hoc networks: A survey. Advanced Computing & Communication Technologies (ACCT), 2012 Second International Conference on, pp. 535-541, Haryana, India.

Johnson, D. B., Maltz, D. A., &, & Broch, J. (2001). DSR: The Dynamic Source Routing Protocol for Multi-hop Wireless Adhoc Networks. In: C. E. Perkins (Ed.), *Adhoc Networking*, pp. 139-172. Addison-Wesley Longman Publishing Co., Boston, USA.

Kannhavong, B., Nakayama, H., Nemoto, Y., Kato, N., & Jamalipour, A. (2007). A Survey of Routing Attacks in Mobile Adhoc Networks. Wireless Communications, IEEE, 14(5), 85-91.

Kumar, S. J., & Saraswathi, R. (2012). A Combined Credit Risk and Collaborative Watchdog Method for Detecting Selfish Node over Mobile Adhoc Network. *International Journal of Advanced Research in Computer Science and Software Engineering, 2*(11), 265-273.

Liu, H., Delgado-Frias, J. G., & , & Medidi, S. (2007). Using a Cache Scheme to Detect Selfish Nodes in Mobile Adhoc Networks. Proceedings of the 6th International Conference on Communications, Internet and Information Technology.

Liu, H., Delgado-Frias José G., & S., M. (2007). Using a Two-Timer Scheme To Detect Selfish Nodes In Mobile Adhoc Networks. Proceedings of the 6th International Conference on Communications, Internet and Information Technology.

Liu, K. (2006). Detecting Routing Misbehavior In Mobile Ad Hoc Network. M.S. Thesis, University of New Orleans.

Liu, K., Deng, J., Varshney, P. K., & Balakrishnan, K. (2007). An Acknowledgment-Based Approach for the Detection of Routing Misbehavior in MANETs. Mobile Computing, IEEE Transactions on, 6(5), 536-550.

Meeneghan, P., & Delaney, D. (2004). An Introduction to NS, Nam and OTcl Scripting. Technical Report Series NUIM-CS-TR-2004, Vol. 5, pp. 1-39. National University of Ireland, Maynooth.

Mohapatra, P., & Krishnamurthy, S. (2005). *Adhoc Networks: Technologies and Protocols*: Springer.

Moses, G. J., Kumar, D. S., Varma, P. S., & Supriya, N. (2012). A Simulation Based Study of AODV, DSR, DSDV Routing Protocols in MANET Using NS-2. *International Journal of Advanced Research in Computer Science and Software Engineering, 2*(3), 43-51.

Padiya, S., Pandit, R., & Patel, S. (2013). Survey of Innovated Techniques to Detect Selfish Nodes in MANET. *International Journal of Computer Networking , Wireless and Mobile Communications,, 3*(1), 221-230.

Rathod, R., Ingle, M., Kankate, B. S., & Kawale, R. (2013). Detection of Routing Misbehaving Links in MANET by 2ACK Scheme. *International Journal of Emerging Technology and Advanced Engineering, 3*(2), 622-626.

Rebahi, Y., Mujica-V, V. E., & Sisalem, D. (2005). A Reputation Based Trust Mechanism for Ad Hoc Networks. Proceedings of the 10th IEEE Symposium on Computers and Communications, pp. 37-42.

Sarkar, S. K., Basavaraju, T., & Puttamadappa, C. (2007). *Adhoc Mobile Wireless Networks: Principles, Protocols and Applications*. CRC Press.

Stallings, W. (2007). *Network Security Essentials: Applications and Standards.* Pearson Education India.

Sun, H.- M., Chen, C.-H., & Ku, Y.-F. (2012). A Novel Acknowledgment Based Approach against Collude Attacks in MANET. *Expert Systems with Applications, 39*(9), 7968-7975.

Tamilarasan, S., & Aramudan, D. (2011). A Performance and Analysis of Misbehaving node in MANET using Intrusion Detection System. *International Journal of Computer Science and Network Security, 11*(5), 258-264.

Tamilarasi, M., & Sundararajan, T. (2012). Secure Enhancement Scheme for Detecting Selfish Nodes in MANET. Computing, Communication and Applications (ICCCA), 2012 International Conference on, pp. 1-5, Tamilnadu, India.

Ukey, A. S. A., & Chawla, M. (2010). Detection of Packet Dropping Attack Using Improved Acknowledgement Based Scheme in MANET. *International Journal of Computer Science Issues, 7*(4), 12-17.

Usha, S., & Radha, S. (2011). Multi Hop Acknowledgement Scheme Based Selfish Node Detection in Mobile Adhoc Networks. *International Journal of Computer and Electrical Engineering, 3*(4), 524-528.

Vijaya, K. (2008). Secure 2ACK Routing Protocol in Mobile Ad Hoc Networks. TENCON 2008-2008 IEEE Region 10 Conference, Hyderabad, India.

Wang, Y., Giruka, V. C., & Singhal, M. (2008). Truthful Multipath Routing for Ad Hoc Networks with Selfish Nodes. *Journal of Parallel and Distributed Computing, 68*(6), 778-789.

Wang, Y., & Singhal, M. (2007). On Improving the Efficiency of Truthful Routing in MANETs with Selfish Nodes. *Pervasive and Mobile Computing, 3*(5), 537-559.

Wang, J. NS-2 Tutorial Exercise. Homepage, < http://www.cs.virginia.edu/~cs757/slidespdf/cs757-ns2-tutorial-exercise.pdf > Accessed 2014, June 8.

Wankhade, S. V. (2012). 2ACK-Scheme: Routing Misbehavior Detection in MANETs Using OLSR. *International Journal of Science, Engineering and Technology Research, 1*(1), 1-7.

Wu, B., Chen, J., Wu, J., & Cardei, M. (2007). A Survey of Attacks and Countermeasures in Mobile Ad Hoc Networks. *Wireless Network Security - Signals and Communication Technology*, pp. 103-135, Springer.

Wu, L.-W., & Yu, R.-F. (2010). A Threshold-Based Method for Selfish Nodes Detection in MANET. Computer Symposium (ICS), 2010 International, pp. - 875-882, Tainan, Southern Taiwan.