Enhanced Adaptive Acknowledgment in MANET'S with Clustering

L. Anita Elizabeth, S. Hari Prasanth, S. Gopesh, B. Kripa Sankar
Department of Information Technology
Sri Venkateswara College of Engineering
Sriperumbudur
lanita@svce.ac.in, s.hari4393@gmail.com, gopeshreddy2293@gmail.com, kripaemm@gmail.com

***Abstract*—**The mobility and scalability brought by wireless network made it possible in many applications. Mobile Ad hoc Network (MANET) is one of the most important and unique applications. On the contrary to traditional network architecture, MANET does not require a fixed network infrastructure; every single node works as both a transmitter and a receiver. The self configuring ability of nodes in MANET made it popular among critical mission applications like military use or emergency recovery. However, the open medium and wide distribution of nodes make MANET vulnerable to malicious attackers. In this case, it is crucial to develop efficient intrusion-detection mechanisms to protect MANET from attacks. Our proposed protocol Enhanced Adaptive Acknowledgment (EAACK) is to detect and perform defence mechanism for packet replication attack. The performance of this proposed system is verified using the parameters like Loss, Delay, Throughput, Channel Usage, Protocol Efficiency, Packet-Delivery Ratio, End-to-End Latency and Bandwidth. Compared to contemporary approaches, EAACK demonstrates higher malicious behaviour detection rates in certain circumstances while does not greatly affect the network performances.

Keywords*-Adaptive acknowledgement; EAACK protocol*

*****

I. INTRODUCTION

Recent advancements in wireless communication and the miniaturization of computers have led to a new concept called the mobile ad hoc network (MANET), where two or more mobile nodes can form a temporary network without need of any existing network infrastructure or centralized administration. Nodes communicate directly with each other when they are both within the same communication range. Otherwise, they rely on their neighbors to relay messages. Even if the source and the destination mobile hosts are not in the communication range of each other, data packets are forwarded to the destination mobile host by relaying the transmission through other mobile hosts which exist between the two mobile hosts. Since no special infrastructure is required, in various fields such as military and rescue affairs, many applications are expected to be developed for ad hoc networks. In ad hoc networks, since mobile hosts move freely, disconnections occur frequently, and this causes frequent network partition. If a network is partitioned into two networks due to the migrations of mobile hosts, mobile hosts in one of the partitions cannot access data items held by mobile hosts in the other. Thus, data accessibility in ad hoc networks is lower than that in conventional fixed networks. To adjust to such trend, we strongly believe that it is vital to address its potential issues. In this case, it is crucial to develop efficient intrusion-detection mechanisms to protect MANET from attacks. We also suggest secure way of transferring messages between mobile nodes. [1]

In this paper, an intrusion-detection protocol named Enhanced Adaptive Acknowledgment (EAACK) is proposed specially designed for MANET's using the concept of Hierarchical clustering of nodes.[5] The motivation behind this approach is the locality property, meaning that if a cluster can be established, nodes typically remain within a cluster, only some nodes change clusters. If a topology within a cluster changes, only nodes of the cluster have to be informed. Nodes of other clusters only need to know how to reach the cluster. The approach basically hides all the small details in clusters which are further away.

II. LITERATURE SURVEY

Anantvalee and J. Wu, in their paper "A Survey on Intrusion Detection in Mobile Ad Hoc Networks" [2] proposed many Intrusion Detection System (IDS) like Watchdog, Pathrater, CONFIDANT, CORE and OCEAN. IDS types are usually classified based on anomaly, misuse or specification based. They are either stand alone, distributed, cooperative or hierarchical. A watchdog identifies the misbehaving nodes by eavesdropping on the transmission of the next hop. A Pathrater then helps to find the routes that do not contain those nodes. Watchdog is capable of detecting malicious nodes rather than links. CONFIDANT (Cooperation of Nodes, Fairness In Dynamic Ad-hoc NeT-works), which is similar to Watchdog and Pathrater. Each node observes the behaviors of neighbor nodes within its radio range and learns from them. This system also solves the problem of Watchdog and Pathrater such that misbehavior nodes are punished by not including them in routing and not helping them on forwarding packets. CORE prevents false accusation, thus, it also prevents a denial of service attack, which cannot be done in CONFIDANT. OCEAN can be considered as a stand-alone architecture. It categorizes routing misbehavior into two types: misleading and selfish. If a node has participated in the route discovery but not packet forwarding, this is considered to be misleading as it misleads other nodes to route packets through it. But if a node does not even participate in the route discovery, it is considered to be selfish. While all these detect selfish nodes they don't find malicious behavior of nodes.

Panagiotis Papadimitratos and Zygmunt J. Haas, in their paper "Secure Data Communication in Mobile Ad Hoc Networks" [3] proposed Secure Message Transmission (SMT) and Secure Single Path (SSP) protocols. The salient features of SMT and SSP is their ability to operate solely in an end-to-end

manner and without restrictive assumptions on the network trust and security associations. SMT involves secure data communication but does not provide effective when large number of nodes is connected. SSP urges for a secured single path from source to destination till the delivery of entire set of packets are done. Unlike previous works, SMT provides a solution tailored to MANET environment, combining four elements: 1) reliance only on end-to-end security bindings; 2) simultaneous transmission across multiple, diverse routes determined by the protocol; 3) robust detection of communication faults; and 4) adaptation to the network condition. But this combination increases the overhead further. The security and fault-tolerance of the data communication are paramount in the inherently insecure and unreliable ad hoc networking environments only in conjunction with secure routing protocols such as SRP, SLSP, or QoS-SRP.

Loay Abusalah, Ashfaq Khokhar, and Mohsen Guizani, in their survey "A Survey of Secure Mobile Ad Hoc Routing Protocols" [4] analyzed Ad Hoc on demand Distance Vector routing (AODV), Optimized Link State Routing (OLSR), and Temporally Ordered Routing Algorithm (TORA) protocols. Routing protocols for ad hoc wireless networks can be classified into three types based on the underlying routing information update mechanism employed. An ad hoc routing protocol could be reactive (on demand), proactive (table driven) or hybrid. Reactive routing protocols obtain the necessary path when it is required, by using a connection establishment process. They do not maintain the network topology information and they do not exchange routing information periodically. In proactive routing protocols, such as DSDV, every node maintains the network topology information in the form of routing tables by periodically exchanging routing information. Routing information is generally flooded in the whole network. Whenever a node requires a path to a destination, it runs an appropriate path finding algorithm on the topology information it maintains. One important problem is determining the resources available at any particular node. AODV has huge amount of overhead involved in maintaining a set of table for each node and their partners. Due to constant changing topography these tables will be a problem as they need to be updated regularly.

Jochen H. Schiller, in his book "Mobile Communications" [5] proposed Hierarchical clustering of nodes. If it is possible to identify certain groups of nodes belonging together, clusters can be established. While individual nodes might move faster, the whole cluster can be rather stationary. Routing between clusters might be simpler and less dynamic. Algorithms such as DSDV, AODV, and DSR only work for a smaller number of nodes and depend heavily on the mobility of nodes. For larger networks, clustering of nodes and using different routing algorithms between and within clusters can be a scalable and efficient solution. The motivation behind this approach is the locality property, meaning that if a cluster can be established, nodes typically remain within a cluster, only some change clusters. If a topology within a cluster changes, only nodes of the cluster have to be informed. Nodes of other clusters only need to know how to reach the cluster. The approach basically hides all the small details in clusters which are further away. From time to time each node needs to get some information about the topology. Again, updates from clusters further away will be sent out less frequently compared to local updates. Clusters can be combined to form super clusters etc., building up a larger hierarchy. Using this approach, one or more nodes can act as cluster heads, representing a router for all traffic to/from the cluster. All nodes within the cluster and all other cluster heads use these as gateway for the cluster.

III. EXISTING SYSTEM

A. BACKGROUND

In particular, considering the fact that most routing protocols in MANETs assume that every node in the network behaves cooperatively with other nodes and presumably not malicious, attackers can easily compromise MANETs by inserting malicious or non cooperative nodes into the network. Furthermore, because of MANET's distributed architecture and changing topology, a traditional centralized monitoring technique is no longer feasible in MANETs. In such case, it is crucial to develop an intrusion-detection system (IDS).

Due to the limitations of most MANET routing protocols, nodes in MANETs assume that other nodes always cooperate with each other to relay data. This assumption leaves the attackers with the opportunities to achieve significant impact on the network with just one or two compromised nodes. To address this problem, Intrusion Detection System should be added to enhance the security level of MANETs. If MANET can detect the attackers as soon as they enter the network, we will be able to completely eliminate the potential damages caused by compromised nodes at the first time. IDS's usually act as the second layer in MANETs, and they are a great complement to existing proactive approaches.

B. WATCHDOG AND PATHRATER

Watchdog aims to improve the throughput of network with the presence of malicious nodes. Watchdog detects malicious misbehaviors by promiscuously listening to its next hop's transmission. If a Watchdog node overhears that its next node fails to forward the packet within a certain period of time, it increases its failure counter. Whenever a node's failure counter exceeds a predefined threshold, the Watchdog node reports it as misbehaving. In this case, the Pathrater cooperates with the routing protocols to avoid the reported nodes in future transmission. But the Watchdog scheme fails to detect malicious misbehaviors with the presence of the following:

1) Ambiguous collisions
2) Receiver collisions
3) Limited transmission power
4) False misbehavior report
5) Partial dropping
6) Collusions

C. TWOACK

TWOACK is neither an enhancement nor a Watchdog-based scheme. Aiming to resolve the receiver collision and limited transmission power problems of Watchdog, TWOACK detects misbehaving links by acknowledging every data packet transmitted over every three consecutive nodes along the path from the source to the destination. Upon retrieval of a packet, each node along the route is required to send back an acknowledgment packet to the node that is two hops away from it down the route. TWOACK is required to work on routing protocols such as Dynamic Source Routing (DSR). The TWOACK scheme successfully solves the receiver

collision and limited transmission power problems posed by Watchdog. However, the acknowledgment process required in every packet transmission process added a significant amount of unwanted network overhead. Due to the limited battery power nature of MANETs, such redundant transmission process can easily degrade the life span of the entire network.

D. ACKNOWLEDGMENT

In the ACK scheme the source node sends out packet to the destination node. All the intermediate nodes simply forward this packet. When the destination node receives packet, it is required to send back an ACK acknowledgement packet to the source node along the reverse order of the same route. Within a predefined time period, if the source node receives this ACK acknowledgement packet, then the packet transmission from source node to destination node is successful. Otherwise the source node will send the packet again to destination till it receives correct acknowledgement from the destination node which causes extra traffic in network and resulting in substantial delay in entire transfer of packets.

IV. PROPOSED SYSTEM

EAACK consists of three major parts, namely, ACK, Secure ACK (S-ACK), and misbehavior report authentication (MRA). EAACK mainly tries to resolve two problems common in MANET namely receiver collisions and limited transmission problem. Both ACK scheme and TWOACK scheme are vulnerable to false misbehavior attack. Also digital signature scheme is used during the packet transmission process for further security to the system. Fig 4.1 gives a clear description of the entire process of the EAACK protocol.

A. ACK SCHEME

This is same as that of the acknowledgement scheme used in wired networks. It acts as a part of the hybrid scheme in EAACK, aiming to reduce network overhead when no network misbehavior is detected. In ACK mode, source node S first sends out an ACK data packet *P* to the destination node D. If all the intermediate nodes along the route between nodes S and D are cooperative and node D successfully receives *P*, node D is required to send back an ACK acknowledgement packet *P* along the same route but in a reverse order. Within a predefined time period, if node S receives *P*, then the packet transmission from node S to node D is successful. Otherwise, node S will switch to S-ACK mode by sending out an S-ACK data packet to detect the misbehaving nodes in the route.

B. S-ACK SCHEME

The S-ACK scheme is an improved version of the TWOACK scheme. The principle is to let every three consecutive nodes work in a group to detect misbehaving nodes. For every three consecutive nodes in the route, the third node is required to send an S-ACK acknowledgement packet to the first node.

The intention of introducing S-ACK mode is to detect misbehaving nodes in the presence of receiver collision or limited transmission power. Consider three consecutive nodes (i.e., F1, F2, and F3) work in a group to detect misbehaving nodes in the network. Node F1 first sends out S-ACK data packet *Psad*1 to node F2. Then, node F2 forwards this packet to node F3. When node F3 receives *Psad*1, as it is the third node in this three-node group, node F3 is required to send back an S-ACK acknowledgement packet *Psak*1 to node F2. Node F2 forwards *Psak*1 back to node F1. If node F1 does not receive this acknowledgement packet within a predefined time period, both nodes F2 and F3 are reported as malicious. Moreover, a misbehavior report will be generated by node F1 and sent to the source node S. Nevertheless, unlike the TWOACK scheme, where the source node immediately trusts the misbehavior report, EAACK requires the source node to switch to MRA mode and confirm this misbehavior report. This is a vital step to detect false misbehavior report in our proposed scheme.

C. MRA SCHEME

The MRA (Misbehavior Report Authentication) scheme is designed to resolve the weakness of Watchdog when it fails to detect misbehaving nodes with the presence of false misbehavior report. The false misbehavior report can be generated by malicious attackers to falsely report innocent nodes as malicious. This attack can be lethal to the entire network when the attackers break down sufficient nodes and thus cause a network division. The core of MRA scheme is to authenticate whether the destination node has received the reported missing packet through a different route. To initiate the MRA mode, the source node first searches its local knowledge base and seeks for an alternative route to the destination node. If there is no other that exists, the source node starts a DSR routing request to find another route. Due to the nature of MANETs, it is common to find out multiple routes between two nodes. By adopting an alternative route to the destination node, we circumvent the misbehavior reporter node. When the destination node receives an MRA packet, it searches its local knowledge base and compares if the reported packet was received. If it is already received, then it is safe to conclude that this is a false misbehavior report and whoever generated this report is marked as malicious. Otherwise, the misbehavior report is trusted and accepted. By the adoption of MRA scheme, EAACK is capable of detecting malicious nodes despite the existence of false misbehavior report.

D. DIGITAL SIGNATURES

Digital signatures have always been an integral part of cryptography in history. Cryptography is the study of mathematical techniques related to aspects of information security such as confidentiality, data integrity, entity authentication and data origin authentication. The development of cryptography technique has a long and fascinating history. The pursuit of secure communication has been conducted by human being since 4000 years ago in Egypt, according to Kahn's book in 1963. The security in MANETs is defined as a combination of processes, procedures and systems used to ensure confidentiality, authentication, integrity, availability and non repudiation. All three parts of EAACK, namely, ACK, S-ACK, and MRA, are acknowledgement-based detection schemes. They all rely on acknowledgement packets to detect misbehaviors in the network. Thus, it is extremely important to ensure that all acknowledgement packets in EAACK are authentic and untainted. Otherwise, if the attackers are smart enough to forge acknowledgement packets, all of the three schemes will be vulnerable. In order to ensure the integrity of the IDS, EAACK requires all acknowledgement packets to be digitally

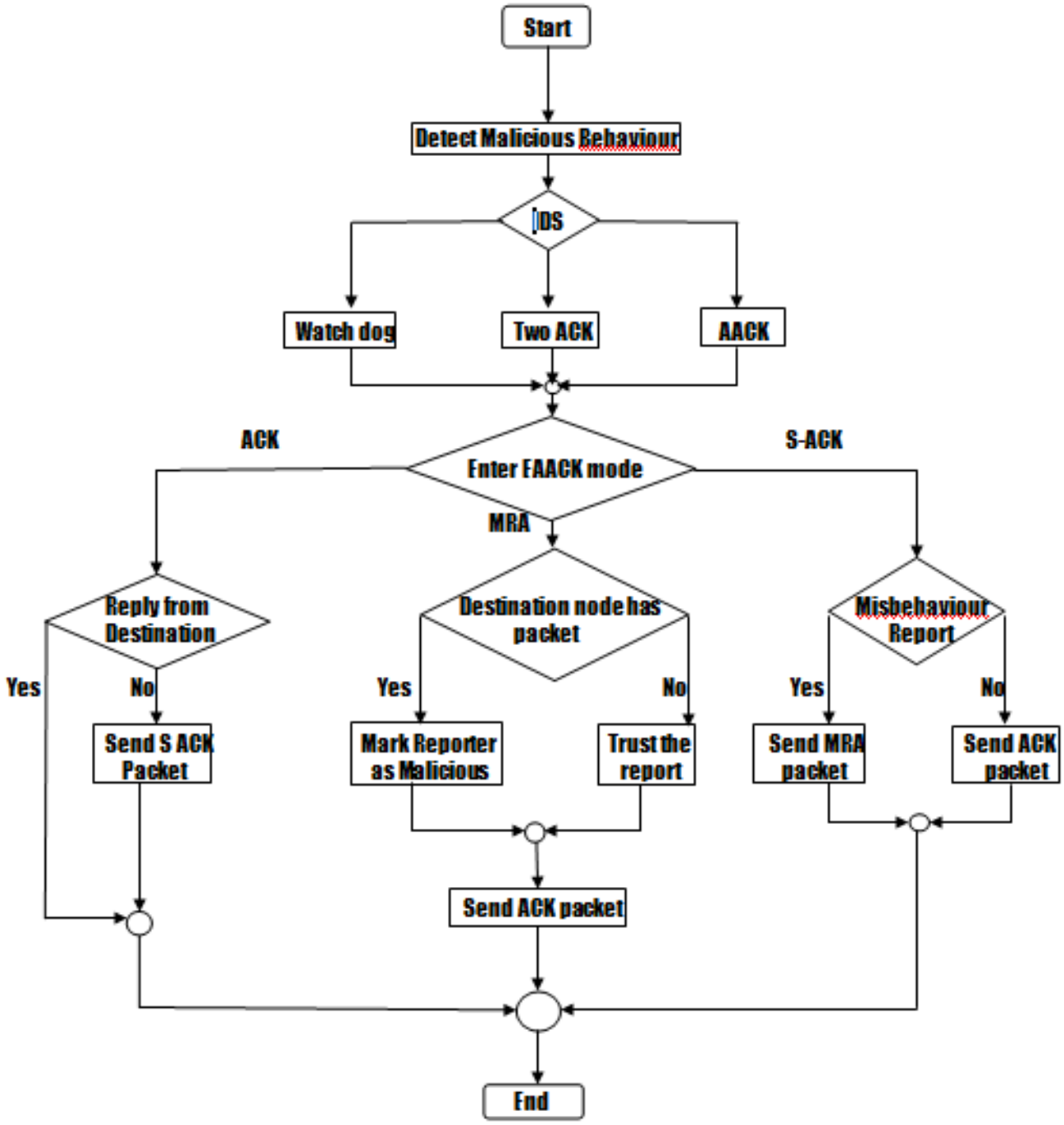

Fig 4.1 System Flow diagram

signed before they are sent out and verified until they are accepted. Following Fig 4.2 is a sample digital signature algorithm commonly referred as RSA algorithm. All three parts of EAACK, namely, ACK, S-ACK, and MRA, are acknowledgement-based detection schemes. They all rely on acknowledgement packets to detect misbehaviors in the network. Thus, it is extremely important to ensure that all acknowledgement packets in EAACK are authentic and untainted.

E. CLUSTERING

Hierarchical clustering of nodes in MANET's is found to be an optimum solution to increase performance when there are more number of nodes present in the network. Algorithms such as DSDV, AODV, and DSR only work for a smaller number of nodes and depend heavily on the mobility of nodes. For larger networks, clustering of nodes and using different routing algorithms between and within clusters can be a scalable and efficient solution. The motivation behind this approach is the locality property, meaning that if a cluster can be established, nodes typically remain within a cluster, only some change clusters. If a topology within a cluster changes, only nodes of the cluster have to be informed. Nodes of other clusters only need to know how to reach the cluster. The approach basically hides all the small details in clusters which are further away. From time to time each node needs to get some information about the topology. Again, updates from

clusters further away will be sent out less frequently compared to local updates. Clusters can be combined to form super clusters etc., building up a larger hierarchy. Using this approach, one or more nodes can act as cluster heads, representing a router for all traffic to/from the cluster. All nodes within the cluster and all other cluster heads use these as gateway for the cluster.

Key Generation

Select p, q	p, q both prime, p≠q
Calculate $n = p \times q$	
Calculate $\phi(n) = (p-1) \times (q-1)$	
Select integer e	$gcd(\phi(n), e) = 1; 1 < e < \phi(n)$
Calculate d	
Public key	$KU = \{e, n\}$
Private key	$KR = \{d, n\}$

Encryption

Plaintext:	$M < n$
Ciphertext:	$C = M^e \pmod n$

Decryption

Ciphertext:	C
Plaintext:	$M = C^d \pmod n$

Fig 4.2 RSA Algorithm

Fig 4.3 shows an ad-hoc network with interconnection to the internet via a base station. This base station transfers data to and from the cluster heads. In this example, one cluster head also acts as head of the super cluster, routing traffic to and from the super cluster. Different routing protocols may be used inside and outside clusters.

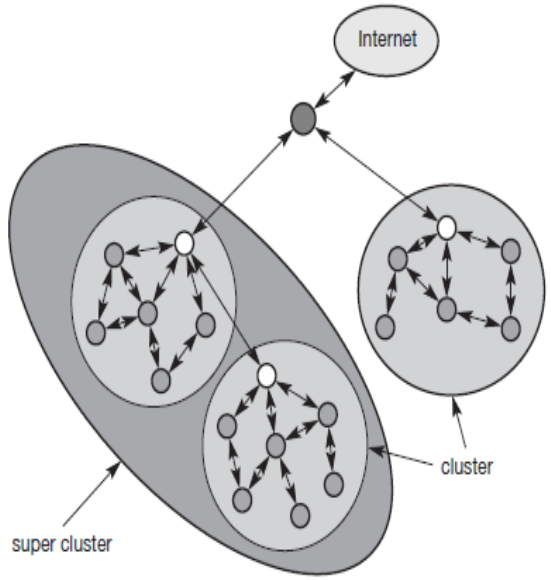

Fig 4.3 Hierarchical clustering

Many hierarchical algorithms are used. They are either Agglomerative or Divisive. Agglomerative is a "bottom up" approach where each node starts its own cluster and pairs of clusters are merged as one moves up the hierarchy. Divisive is a "top down" approach where all observations start in one cluster, and splits are performed recursively as one moves down the hierarchy. This bottom up approach is used in MANET's when clustering is done.

Fig 4.4 shows the representation of progressively merging clusters. One commonly used algorithm is Nearest Neighbor algorithm which is explained below

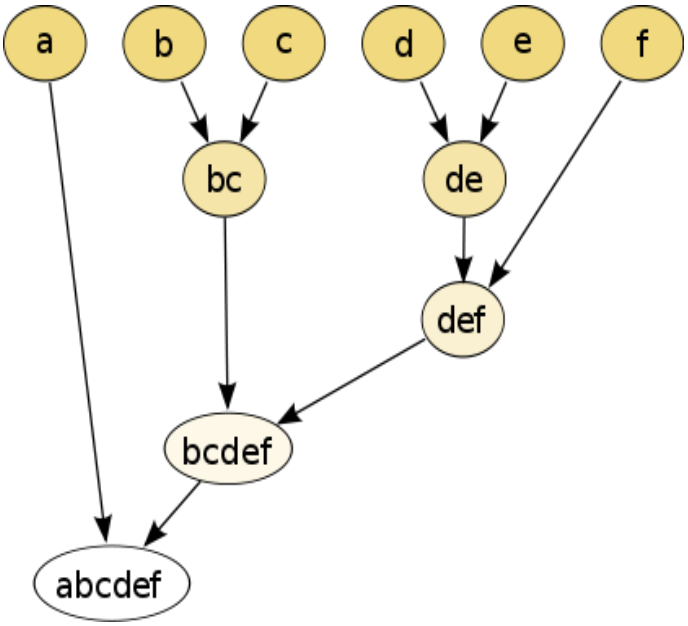

Fig 4.4 Representation of progressively merging clusters

NEAREST-NEIGHBOUR CLUSTERING ALGORITHM

1. **begin**
2. **initialize**
   c; c' = n; $D_i = \{x_i\}$; i = 1,...,n
3. **do**
4. c' = c' - 1
5. Find nearest clusters D_i and D_j
6. Merge D_i and D_j
7. **until**
   c = c'
8. **Return**
   c clusters
9. **End**

Where c= no. of clusters
d= distance between nodes

To find the nearest clusters in step 5, the following clustering criterion function is used:

$$d_{min}(D_i, D_j) = \min \| x - x' \|,$$
where $x \varepsilon D_i$ and $x' \varepsilon D_j$

The merging of the two clusters in step 6 simply corresponds to adding an edge between the nearest pair of nodes in D_i and D_j.

Nodes which are near to each other are found using the distance of separation. Then they are merged using nearest neighbor algorithm. A head node is selected among the

merged nodes group. This is done by taking the node which has the maximum amount of energy. The overhead involved in creating these clusters is over come by the fact that less memory is needed for routing tables. Only the details of nearby nodes are stored. The best path for any nodes travelling between clusters is selected with details stored in head node of each cluster.

V. PERFORMANCE EVALUATION

In this section, we concentrate on describing our simulation environment and methodology as well as comparing performances through simulation result comparison with the existing and the proposed EAACK schemes.

A. Simulation Methodologies

Our simulation is conducted within the Network Simulator 2 (NS2) 2.34 environment on a platform with GCC 4.3 and Ubuntu 9.10. The system is running on a laptop with INTEL i5 Processor CPU and 4-GB RAM.

In order to better compare our simulation results with other research works, we adopted the default scenario settings in NS2.34. The intention is to provide more general results and make it easier for us to compare the results.

For each scheme, we ran every network scenario three times and calculated the average performance. In order to measure and compare the performances of our proposed scheme, we continue to adopt the following performance metrics.

1) *Packet delivery ratio (PDR)*: PDR defines the ratio of the number of packets received by the destination node to the number of packets sent by the source node.

2) *Routing overhead (RO)*: RO defines the ratio of the amount of routing-related transmissions [Route REQuest (RREQ), Route REPly (RREP), Route ERRor (RERR), ACK, S-ACK, and MRA].

3) *Packet Loss:* Defines the number of packets lost during transmission.

4) *Delay:* This defines the delay time for a packet to reach destination node from source node.

5) *Throughput:* This defines the maximum attainable packet delivery ratio.

6) *Channel Utilization:* Defines the maximum way a channel can be utilized.

7) *Protocol Efficiency:* Defines the overall efficiency of EAACK protocol.

All these measures are compared with existing and proposed systems and suitable graph plots are shown. In all the comparison graphs, time is taken in X axis and corresponding specific parameter is taken in Y axis. For traditional system 25 to 30 nodes were taken and for EAACK protocol around 50 nodes were taken. It is concluded that even with comparatively more number of nodes EAACK is found to be highly efficient. The performance is very astonishingly high. But also the overhead in formation of clusters is also to be considered which reduces the performance level. Even with this overhead involved EAACK proves to be best than other protocols of MANETs.

During the simulation, the source route broadcasts an RTS (Request to Send) message to all the neighbors within its communication range. Upon receiving this RTS message, each neighbor appends their addresses to the message and broadcasts this new message to their neighbors. If any node receives the same RTS message more than once, it ignores it. If a failed node is detected, which generally indicates a broken link in flat routing protocols like DSR, a error message is sent to the source node. When the RTS message arrives to its final destination node, the destination node initiates an CTS (Clear to Send) message and sends this message back to the source node by reversing the route in the RTS message.

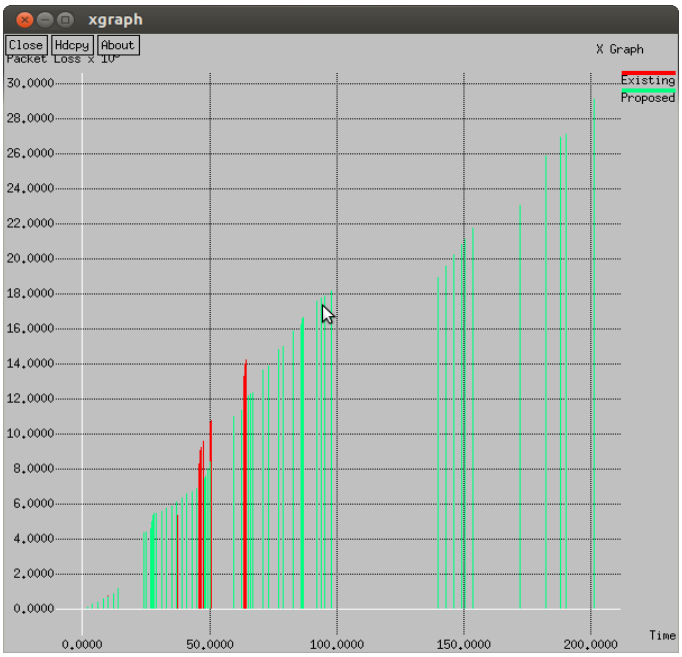

Fig 6.1 Packet loss

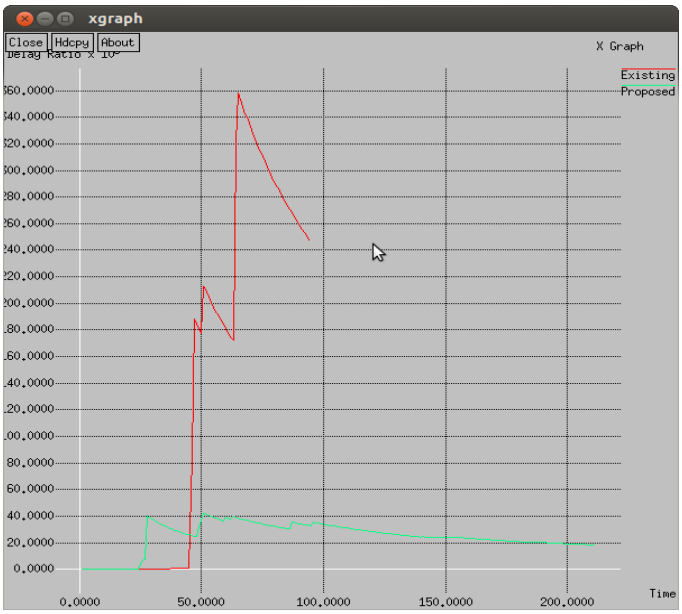

Fig 6.2 Delay

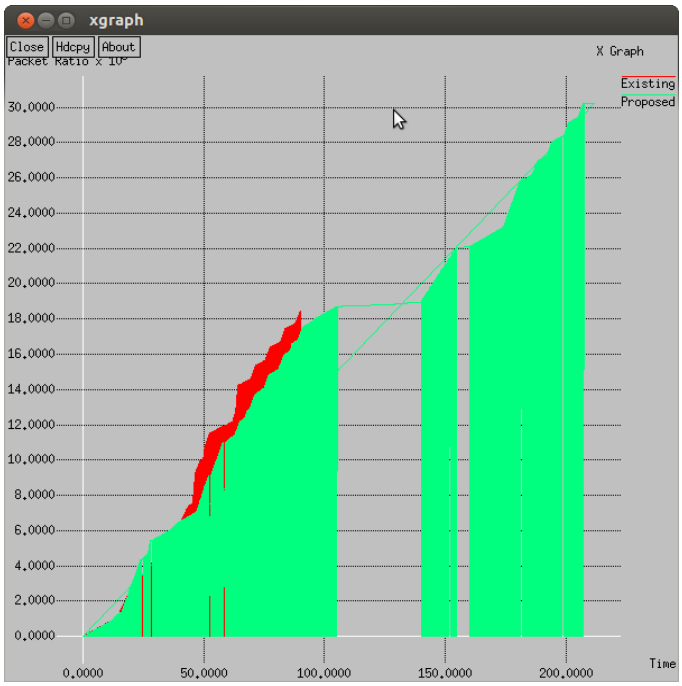

Fig 6.3 Throughput

VI. CONCLUSION AND FUTURE WORK

EAACK protocol is found to be more efficient than other protocols with the use of hybrid cryptography techniques. Even though it generates more routing overheads in some cases it can vastly improve the network's packet delivery ratio. Even with larger network topography the efficiency of this protocol doesn't vary largely.

EAACK protocol performance can be further enhanced by using more efficient clustering algorithm and implementing it in very large scale. Also implementing it in real time, the protocol yields better results than NS2 simulation results.

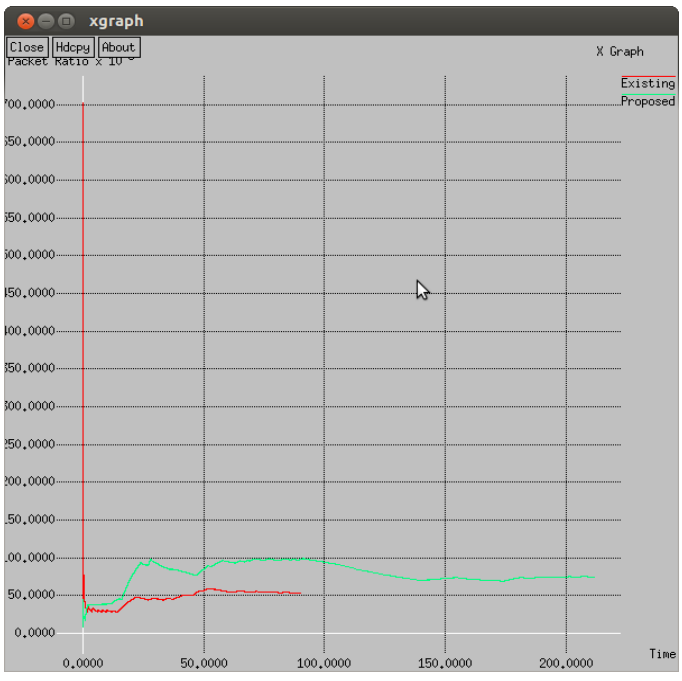

Fig 6.4 Channel Utilization

Fig 6.5 Protocol Efficiency

REFERENCES

[1] Elhadi M. Shakshuki, Nan Kang, and Tarek R. Sheltami, "EAACK—A Secure Intrusion Detection System for MANETs" in IEEE transactions on industrial electronics, vol. 60, no.3, March 2013, pp.1089-1098.

[2] T. Anantvalee and J. Wu, "A Survey on Intrusion Detection in Mobile Ad Hoc Networks," in Wireless/Mobile Security. New York: Springer-Verlag, 2008, pp.170-196.

[3] Panagiotis Papadimitratos and Zygmunt J. Haas, "Secure Data Communication in Mobile Ad Hoc Networks", in IEEE journal on selected areas in communications, vol. 24, No. 2, February 2006, pp.343-356.

[4] Loay Abusalah, Ashfaq Khokhar, and Mohsen Guizani, "A Survey of Secure Mobile Ad Hoc Routing Protocols" in IEEE Communications surveys & tutorials, vol. 10, no. 4, Fourth quarter 2008, pp.78-93.

[5] Jochen H.Schiller, "Mobile Communications", Pearson Education Ltd, Second Edition 2003, ISBN 0321123816, pp-335.