Fidelity Based On Demand Secure(FBOD) Routing in Mobile Adhoc Network

Himadri Nath Saha
Assistant Professor
Department of Computer Science and Engineering,
Institute of Engineering and Management
West Bengal, India.

Dr. Debika Bhattacharyya
Professor
Department of Computer Science and Engineering.
Institute of Engineering and Management, West Bengal, India.

Dr. P. K.Banerjee
Professor
Department of Electronics and Communication Engineering.
Jadavpur University, West Bengal, India.

***Abstract*—: In mobile ad-hoc network (MANET), secure routing is a challenging issue due to its open nature, infrastructure less property and mobility of nodes. Many mobile ad-hoc network routing schemes have been proposed, but none of them have been designed with security as a goal. We propose security goals for routing in mobile ad-hoc networks, an approach significantly different from the existing ones where data packets are routed, based on a specific criterion of the nodes called "fidelity" The approach will reduce the computational overhead to a lot extent. Our simulation results show how we have reduced the amount of network activity for each node required to route a data packet and how this scheme prevents various attacks which may jeopardize any MANET.**

Keywords- fidelity; sequence number; hop destination; flooding attack; black hole attack; co-operative black hole attac,routing.

I. INTRODUCTION

Mobile Ad-hoc Network (MANET) is a collection of wireless mobile hosts without fixed network [20] infrastructure and centralized administration (Figure-1). Communication in MANET [8] is done via multi-hop paths. MANET contains diverse resources and nodes operate in shared wireless medium. [21] Network topology changes unpredictably and very dynamically. Radio link [31] reliability is necessary as connection breaks are pretty frequent. Moreover, density of nodes, number of nodes and mobility of these hosts may vary in different applications. There is no stationary infrastructure. Each node in MANET [28] acts a router that forwards data packets to other nodes. Therefore selection of effective, suitable, adaptive and robust routing protocol is of utmost importance.

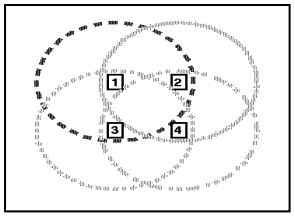

Figure 1: An ad-hoc mobile network with four nodes.

Rest of the paper is organized as follows. We have discussed related work in section 2 and describe the Fidelity in section 3,description of the scheme in section 4,algorithm of proposed scheme in section 5,simulation results in section 6 ,security aspects in section 7, the simulation analysis and performance metrics in section 8 and finally present our conclusions in section 9.

II. RELATED WORK

S. Matri [33] proposed to trace malicious nodes by using watchdog/pathrater. In watchdog when a node forwards a packet, the node's watchdog verifies that the next node in the path also forwards the packet by promiscuously listening to the next node's transmissions. If the watchdog finds the next node does not forward the packet during a predefined threshold time, the watchdog will accuse the next node as a malicious node to the source node; The proposal has two shortcomings: 1) to monitor the behavior of nodes two or more hops away, one node has to trust the information from other nodes, which introduces the vulnerability that good nodes may be bypassed by malicious accusation; 2) The *watchdog* cannot differentiate the misbehavior from the ambiguous collisions, receiver collisions, controlled transmission power, collusion, false misbehavior and partial dropping. In pathrater algorithm each node uses the watchdog's monitored results to rate its one-hop neighbors. Further the nodes exchange their ratings, so that the pathrater can rate the paths and choose a path with highest rating for routing. Shortcoming of this algorithm is that the idea of exchanging ratings genuinely opens door for blackmail attack.

SCAN [11] exploits two ideas to protect the mobile Ad Hoc networks [17]: 1) local collaboration: the neighboring nodes collectively monitor each other and sustain each other; and 2) information cross-validation: each node monitors its neighbors by cross-checking the overheard transmissions, and the monitoring results from different nodes are further cross validated. As a result, the security solution is self-organized, distributed, and fully localized. In SCAN once a malicious node is convicted by its neighbors, the network reacts by depriving its right to access the network by revoking its token. A powerful collusion among the attackers will break SCAN as it violates the assumption of the polynomial secret sharing scheme.

Gonzalez [24] presents a methodology, for detecting packet forwarding misbehavior, which is based on the principle of flow conservation in a network. It states that if all neighbors of a node v_j are queried for i) the amount of packets sent to v_j for forwarding and ii) the amount of packets forwarded by v_j to them, then the total amount of packets sent to and received from v_j must be equal. They assume a threshold value for non

malicious packet drop. A node v_i maintains a table with two metrics T_{ij} and R_{ij}, which contains an entry for each node v_j to which v_i has respectively transmitted packets to or received packets from. Node v_i increments T_{ij} on successful transmission of a packet to v_j for v_j to forward to another node, and increments R_{ij} on successful receipt of a packet forwarded by v_j that did not originate at v_j. All nodes in the network continuously monitor their neighbors and update the list of those they have heard recently. This algorithm does not require many nodes to overhear each others' received and transmitted packets, but instead it uses statistics accumulated by each node as it transmits to and receives data from its neighbors. Since there is no collaborative consensus mechanism, such an algorithm may lead to false accusations against correctly behaving nodes.

Himadri [34, 35, 36], in their literatures have shown ways to mitigate attacks on different MANET networks. We have extended their works n this field.

III. FIDELITY

Fidelity is the most important concept of this routing protocol. Fidelity is an integer number that is associated with each node. This fidelity of a node denotes many things about the node itself and also deciphers other information regarding the topology of the entire network. It also helps to maintain security [29] to some extent.

To make it understandable in one sentence, "fidelity is a counter that is associated with a node, which is increased whenever it forwards a data packet successfully." Whenever a node comes in a network its fidelity is zero and whenever it goes permanently off from the network its value is again refreshed to zero. Otherwise whenever a node will forward any data packet it will always increase a counter value and that counter value is its fidelity. Note whenever a source node sends a data packet to a destination node, all the intermediate nodes helping to transmit its data packet will increase their counter but the source and the destination node do not increase their fidelity value.

Fidelity is a measure of these two factors:-

A. How reliable a node is for forwarding a data packet

Whenever we observe that the fidelity value of a particular node is greater that of another node then we can conclude that the one having the greater value is a more durable node than the other from who's its value is greater. It is quite logical because a node with greater value indicates that it is an experienced node in the network and it has transmitted packets most dutifully than other nodes.

B. Network topology

If we can find some nodes with higher fidelity in a region of the network, we conclude that the network activity is higher in that region. More precisely we can also infer that the node density is also higher in that region for it is impossible to have one node having very high fidelity [19] surrounded by nodes with low fidelity because a high fidelity [18] node must send packets to someone in its vicinity which will make that other node's fidelity value also high. Thus a high fidelity value accounts for high network activity as well as high density of nodes in its surroundings.

IV. DESCRIPTION OF THE SCHEME

The term "friends of a node" used in this paper, indicates actually the nodes that fall in the physical range of a particular node. When nodes are having messages to send, all the nodes will check which nodes are in its neighborhood and they will broadcast a request. After getting reply they will make their friend list. More precisely the friend list consists of a table that contains two attributes. The first one is the address [14] of the nodes which are within its range and other is the fidelity value of that particular node. When each node is updated then they will sort that table according to the decreasing order of the fidelity value. Before we enter into the detailed discussion of our protocol there are some concepts that need to be understood. These are as follows-

There will be a sequence counter in every node. If a message is generated in a node then it will be increased by one. This sequence no. will be forwarded as a part of the message. Every node will maintain a buffer where (source, sequence no.) will be stored for last n no. of received messages. After getting a message a node will verify the tuple [24] (source, sequence no) of that message with those tuples in its buffer [13]. If anyone of them matches with that message then that node will reject that message silently. It will prevent flooding attack.

The timeout period of every node through which message is traversed, will be gradually decreased by a critical factor [15] i.e. if timeout period of sender node is x then timeout period of receiver node will be x/m, where m will be critical factor. This factor [23] signifies maximum no of failure a node can endure without causing congestion in the network.

Now the protocol is as follows-

A node can do either of three activities - message generate, message forward, message receive. If it is not doing any of the three then it is idle. Now if a message is generated in a node and it needs to be sent then the node will remain busy until an acknowledgement is received for this message. It is to be noted that a busy node can accept & process an acknowledgement and can send a fail message.

Now if destination is directly reachable from generator node then it will send message to destination node and will wait for acknowledgement, and remain busy until acknowledgement is received. If the destination node is busy it will send a fail message to generator node. After getting fail message or if timeout period exceeds, generator node will keep on sending the message after a certain time periodically until acknowledgement is received.

If destination is not directly reachable then generator node will send message to the node in its range that has highest fidelity value. If generator node get a fail message from that node or if timeout period exceeds then it will send the message to the node having second highest fidelity value and it will continue like this. If the whole list is exhausted in this way then the process will again continue from the node having highest fidelity value. Only generator node will follow this process.

Other nodes will send a fail message to its predecessor if the whole list is exhausted.

When a node receives a message, if it is busy then it will send a fail message to sender, otherwise it will check whether it itself a destination or not. If it is destination, it will accept the message and send acknowledgement to sender otherwise this node will send message to the node in its range that have highest fidelity value and that process will continue. In that acknowledgement message the sequence no. will be same as received message but source will be substituted by destination.

V. ALGORITHMS

Update friend list

STEP 1: Send broadcast request for friends to reply
STEP 2: Receive replies from neighbours
STEP 3: Update my friend list
STEP 4: Sort friend list

Generated data

STEP 1: Set my status=busy
STEP 2: If destination directly reachable from here
- Send packet to destination
- Wait for ACK
- If ACK received consider success
- Else if timeout occurs or FAIL received, arrange for resending
  Else
- Send data packet to the friend having highest fidelity value
- Wait for ACK
- If ACK received consider success and go to last step
- Else if timeout occurs or FAIL received, arrange for resending to the friend with next highest fidelity value
- Continue above three steps until ACK received
- If list is exhausted without getting an ACK then again start from the friend with the highest fidelity value and try each node in friend list in the manner told above.
- While trying to send if the list is exhausted thrice abort

STEP 3: Set my status=free

Received data

STEP 1: If my status=busy send FAIL to sender

STEP 2: Else
- Make my status=busy
- Process received data
- Make my status=free

Process received data

STEP 1: If message destination=my address
- Accept data
- Generate ACK
- Send the ACK to the node from which it directly received the message

STEP 2: Else
- Forward data packet
- Check if forward operation is successful
- If successful increase my fidelity value by 1 and send ACK to the node from which it directly received the message
- Else send FAIL to the node from which it directly received the message

Forward data packet

STEP 1: If message destination is directly reachable from here
- Send packet to destination
- Wait for ACK
- If ACK received consider success
- Else if timeout occurs or FAIL received, arrange for resending to destination.
- If resending fails 3 times consider failure.

STEP 2: Else
- Send data packet to the friend having highest fidelity value
- Wait for ACK
- If ACK received consider success
- Else if timeout occurs or FAIL received, arrange for resending to the friend with next highest fidelity value
- Continue above three steps until ACK received
- If list is exhausted without getting an ACK then consider failure.

VI. SIMULATION RESULT

We have simulated this protocol with JAVA. We need to know something to make out these simulations. These are-

1. Small circle signifies node in the network.
2. Blue circle around node signifies range of that node.
3. Red color indicates that the node is free.
4. Black color indicates that the node is busy.
5. Yellow line indicates probing for neighbors.
6. Pink line indicates reply of probing.
7. Red line between two nodes indicates sending of message.
8. Green line between two nodes indicates sending of acknowledgement.
9. Blue line between two nodes indicates sending of fail message.
10. Any node inside the range of a node is its neighbor node.

Now we will describe one test case simulation.

This is a network having four nodes. Their corresponding fidelity values are written beside the nodes. Here we are trying to send a message from node 0 to node 3. This is basically a worst case scenario according to our protocol. We will see after sending the message a no of times how our protocol makes this worst case scenario to a best case one.

The design of network is

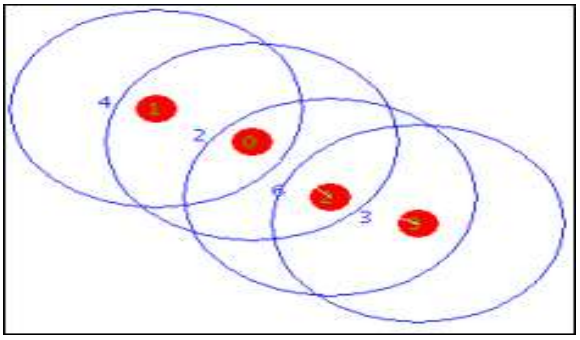

Figure 2: Design of network.

The result we get after net designing is given below-

4 <no of nodes>
2 4 2 3
-1 0 0 -1
0 -1 -1 -1
0 -1 -1 0
-1 -1 0 -1

we edit the adjacency list.txt as:-
4
2 4 2 3
-1 0 0 -1
0 -1 -1 -1
0 -1 -1 0
-1 -1 0 -1
0 <time interval>
0 3 hello <source> <destination> <msg>
10 <time interval>
0 3 hello1 <source> <destination> <msg>
10 <time interval>
0 3 hello2 <source> <destination> <msg>
10 <time interval>
0 3 hello3 <source> <destination> <msg>

then we run the simulation and see the results.

The steps of the visual simulation are given below-

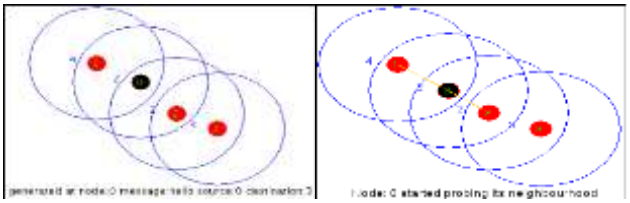

Figure 3: Message generated at node 0. . (left fig.)
Figure 4: Node 0 started probing. (right fig.)

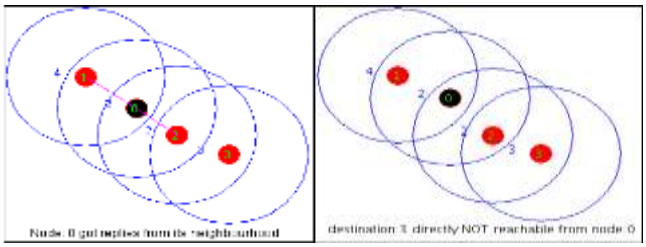

Figure 5: Node 0 got replies from neighbour nodes. . (left fig.)
Figure 6: Destination is not directly reachable from source node. (right fig.)

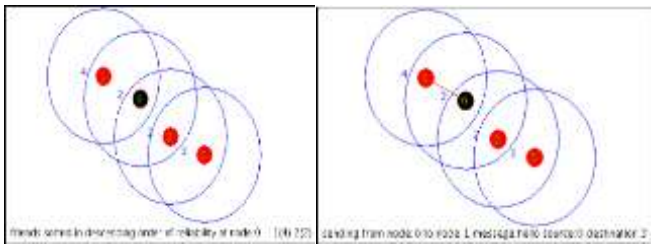

Figure 7: Friend nodes are sorted in descending order. . (left fig.)
Figure 8: Node 0 is sending message to node 1(right fig.)

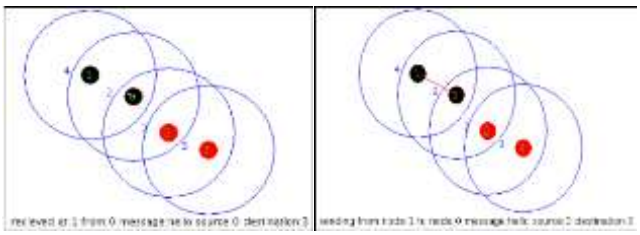

Fig 9: Message is received by node 1 (left fig.)
Fig 10: Node 1 is trying to send message to node 0. (right fig.)

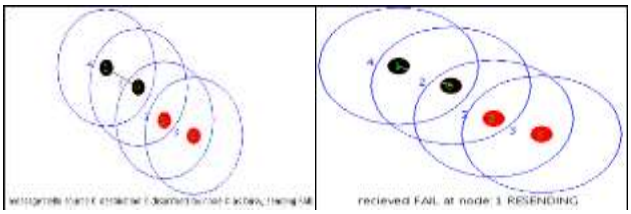

Fig 11: Node 0 discarded the message. . (left fig.)
Fig 12: Node 1 is resending the message. (right fig.)

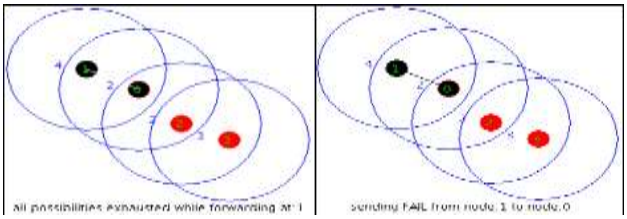

Fig 13: No possible ways to send the message. (left fig.)
Fig 14: Message sending fail from node 1 to node 0. (right fig.)

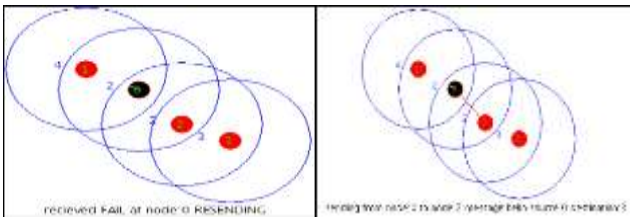

Fig 15: Node 0 resending the message via another path. . (left fig.)
Fig 16: Node 0 sending message to node 2 (right fig.)

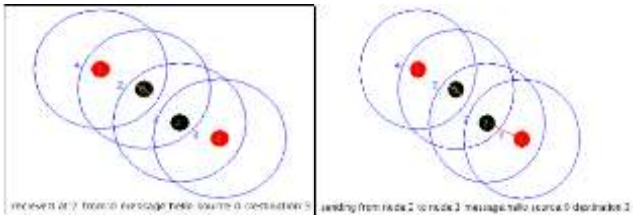

Figure 17: Message received by node 2. . (left fig.)
Figure 18: Node 2 is sending message to node 3. (right fig.)

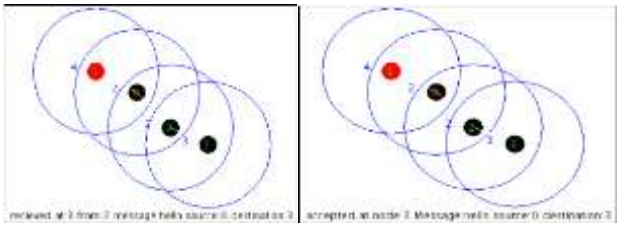

Figure 19: Message received by node 3. (left fig.)
Figure 20: Node 3 accepts the message. . (right fig.)

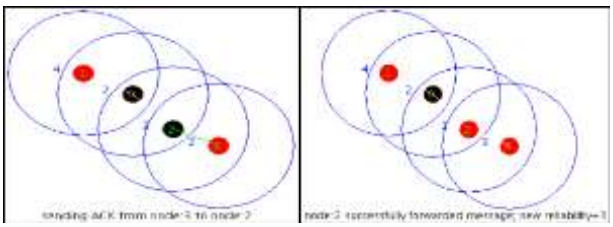

Figure 21: Node 3 is sending ACK to node 2. (left fig.)
Figure 22: Fidelity value of node 2 increases. (right fig.)

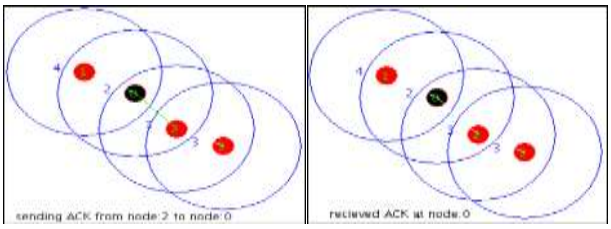

Figure 23: Node 2 is sending ACK to node 0. (left fig.)
Figure 24: Node 0 receives ACK. (right fig.)

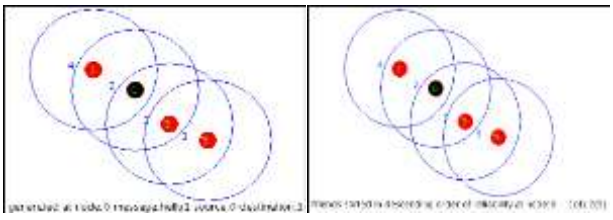

Figure 25: Node 0 wants to send another message to node 3. . (left fig.)
Figure 26: Friend are sorted by node 0 according reliability. (right fig.)

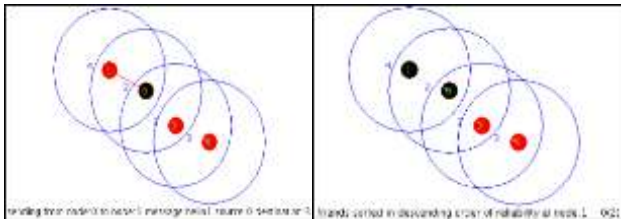

Figure 27: Node 0 is sending message to node 1. . (left fig.)
Figure 28: Friends are sorted by Node 1. (right fig.)

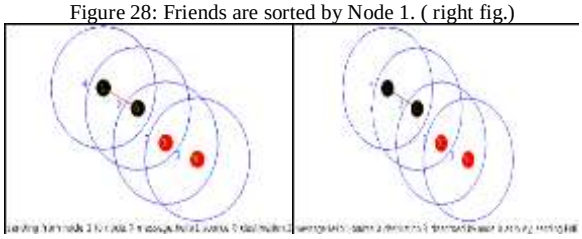

Figure 29: Node 1 is sending message to node 0. . (left fig.)
Figure 30: Node 0 discards the message. (right fig.)

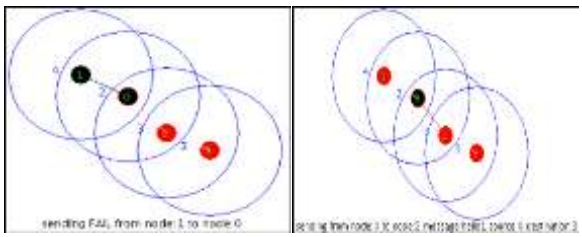

Figure 31: Node 1 fails to send the message. . (left fig.)
Figure 32: Node 0 sends the message to node 2. (right fig.)

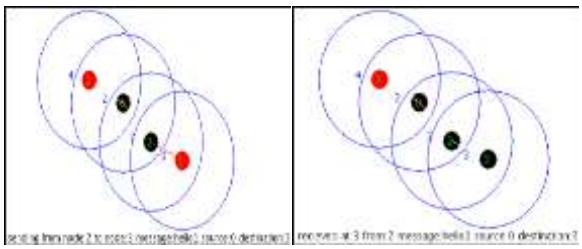

Figure 33: Node 2 sends the messages to node 3. . (left fig.)
Figure 34: Message received by node 3. (right fig.)

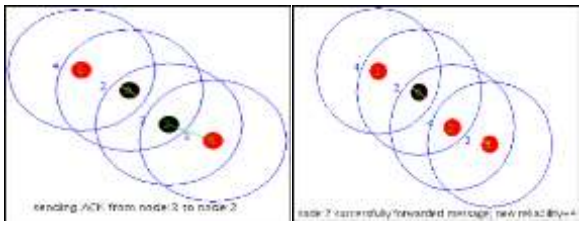

Figure 35: Node 3 is sending ACK to node 2. . (left fig.)
Figure 36: The fidelity value of node 2 increases to 4. (right fig.)

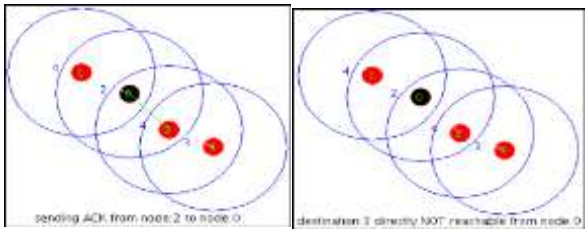

Figure 37: Node 2 sends ACK to node 0. . (left fig.)
Figure 38: Destination unreachable from source (right fig.)

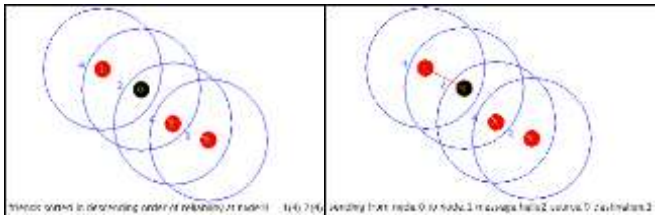

Figure 39: Node 0 sorts its friends to send another message. . (left fig.)
Figure 40: Node 0 is sending message to node 1. (right fig.)

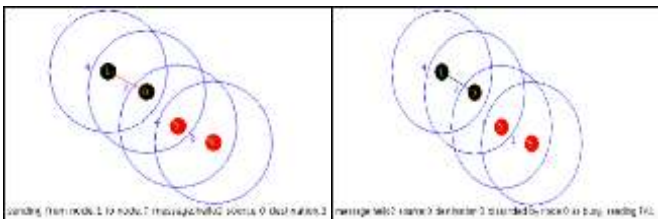

Figure 41: Node 1 is sending message to node 0. . (left fig.)
Figure 42: Node 0 discards the message. (right fig.)

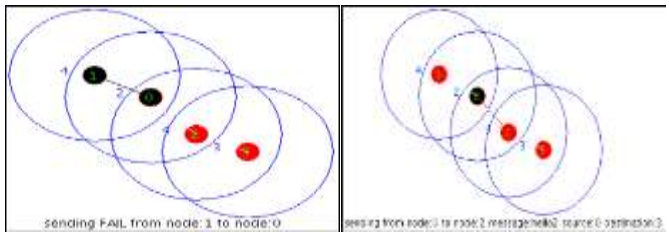

Figure 43: Node 1 couldn't send the message. . (left fig.)
Figure 44: Node 0 sends the message to node 2. (right fig.)

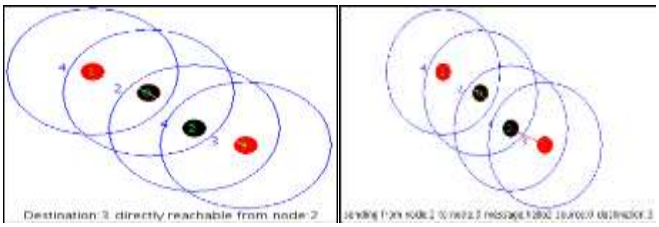

Figure 45: Node to can send the message to destination node. . (left fig.)
Figure 46: Node 2 sending message to node 3. (right fig.)

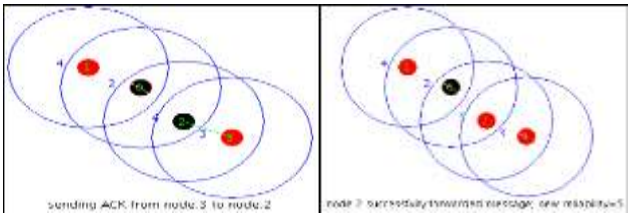

Figure 47: Node 3 sends ACK to node 2. . (left fig.)
Figure 48: Reliability of node 2 increased. (right fig.)

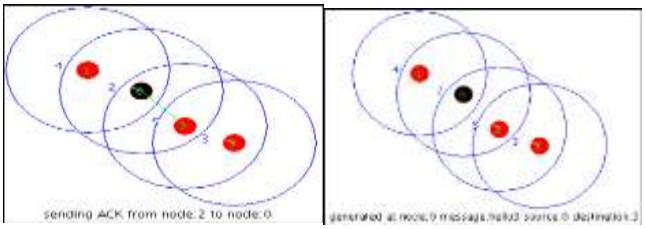

Figure 49: Node 2 sends ACK to node 0. . (left fig.)
Figure 50: New message is generated at node 0. (right fig.)

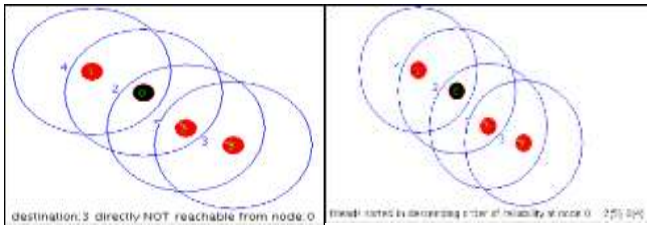

Figure 51: Source and destination node is not directly connected. . (left fig.)
Figure 52: Friends are sorted in descending order at node 0. (right fig.)

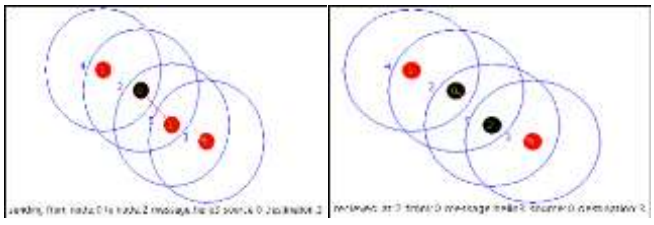

Figure 53: Node 0 sending message to node 2. . (left fig.)
Figure 54: Message received by node 2. (right fig.)

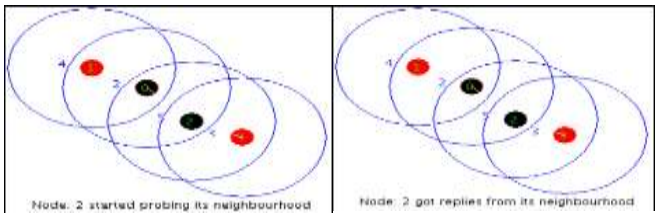

Figure 55: Node 2 started probing its neighbour nodes. . (left fig.)
Figure 56: Node 2 receives reply from neighbours. (right fig.)

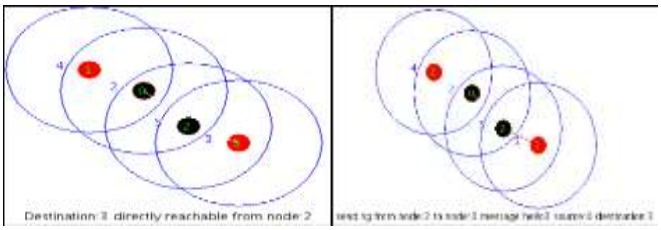

Figure 57: Destination node directly reachable from node 2. . (left fig.)
Figure 58: Node 2 sends message to node 3. (right fig.)

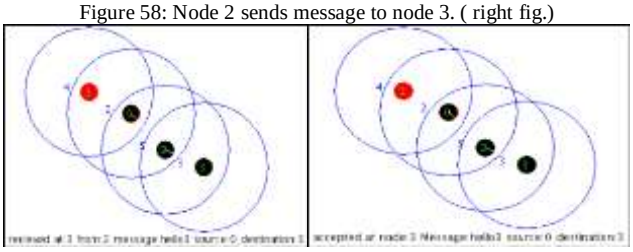

Figure 59: Message reached to node 3. . (left fig.)
Figure 60: Node 3 accepts the message. (right fig.)

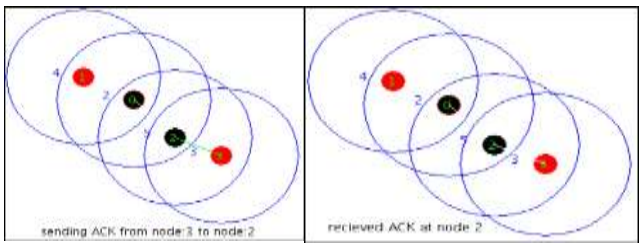

Figure 61: Node 3 sending ACK to node 2. . (left fig.)
Figure 62: Node 2 receives ACK. (right fig.)

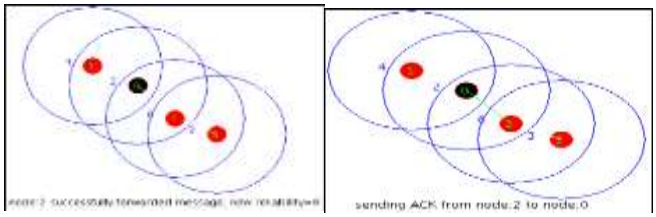

Figure 63: Message successfully forwarded by node 2. . (left fig.)
Figure 64: Node 2 sending ACK to node 0. (right fig.)

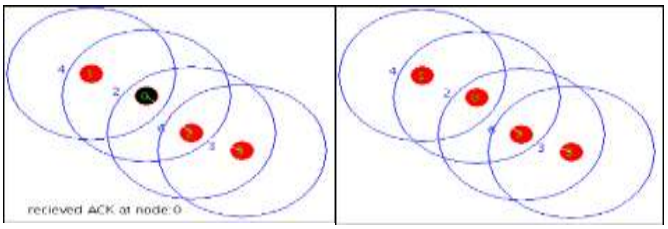

Figure 65: ACK received by node 0. . (left fig.)
Figure 66: All messages transferred successfully. (right fig.)

Message transfer is completed.

VII. Security Aspects

This scheme can efficiently mitigate Flooding attack [3], Black Holes [04] [30], Co-operative Black hole [04], Grey hole [03], Black mail attack [03], Rushing attack [01] and Wormhole Attack [03]. Our simulation has effectively depicted its immunity towards these attacks. This scheme is also safe from attacks to which AODV [08] [30], DSDV [1] is commonly subjected.

VIII. Simulation Analysis And Performance Metrics

In order to evaluate the performance of Ad Hoc network routing protocols, the following matrices were considered:

A. Packet Delivery Fraction

PDF is defined as the ratio between no. of packets originated by application layer [26] in the source node to the no of packets received by the destination node. It will describe the loss rate that will be seen by the transport protocols, which in turn affect the maximum throughput that the network supports. In terms of packet delivery fraction, our protocol FBRP performs well. As the no of nodes getting increased the no packets generated is higher so it may not transfer some of the packets, but the no of these packets are very small. When the no. of nodes is small then in ideal case PDF value is 1. But in case of DSR [10] the PDF is very fluctuating it is lesser in some of the points with respect to the other protocols but it is very higher in some of the points which are not tolerable. DSDV [12] is better in more no. of nodes but AODV [7] [2] is better in smaller no. of nodes region.

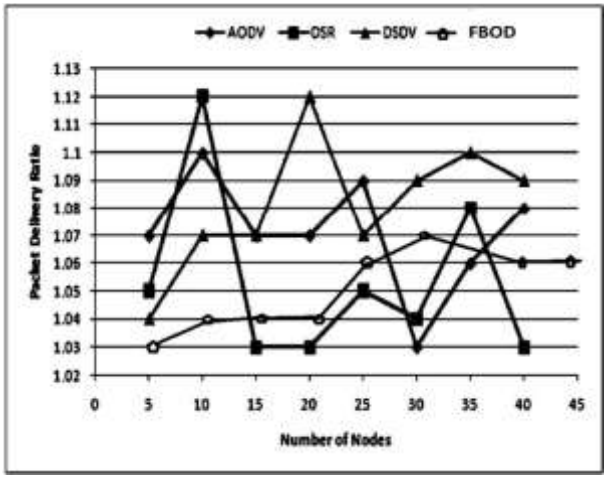

Figure 68.1: Packet Delivery Ratio for AODV, DSR, DSDV, FBOD

B. End to End Delay

The delay is affected by high rate of CBR Packets as well as the buffers become full much quicker, so packets have to stay in the buffer for a longer period of time before they are sent. This can be seen in DSR [8] when it reaches around 2300 packets in 0 mobility. For average end to end delay, the performance of DSR [9] decreases and varies with the number of nodes. In our protocol that is in FBRP the delay is getting increased with the increased no of nodes as the congestion is getting increased. But the rate of this increment id lesser as we don't maintain any kind of buffer. The performance of DSDV [9] id degrading due to increase in the number of nodes the load of exchange of routing tables becomes high and the frequency of exchange also increased. Due to the mobility of nodes the performance of AODV [6] decreases and remains constant as the no of nodes increases.

C. Number of Packets Dropped

The number of data packets that are not successfully sent to the destination is the no of packets being dropped. In terms of dropped packets AODV's [8] performance is the worst. The performance decreases with the increase in the number of packets.

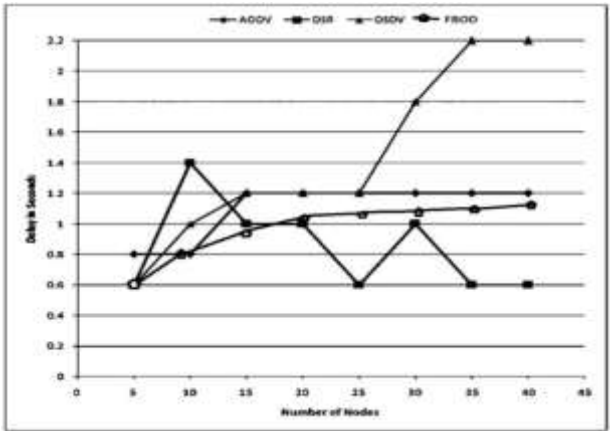

Figure 68.2: Average End to End Delay Ratio for AODV, DSR, DSDV, FBOD

DSDV [8] [9] performs consistently well with increase in the no. of nodes. DSR [10] [9] performs well when no of nodes is less but fails slightly when no of nodes is increased. In our protocol also in ideal case there is no drop of packets with the increase in no of nodes. It performs consistently well.

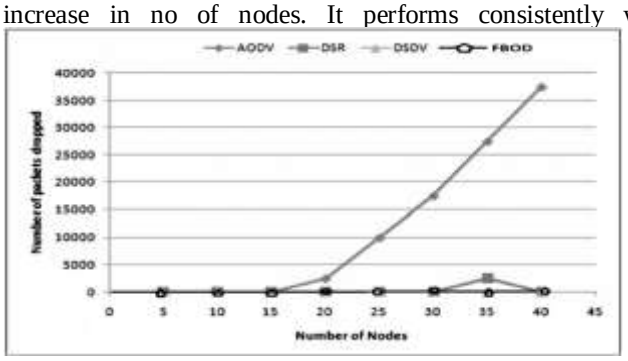

Figure 68.3: Dropped Packets for AODV, DSR, DSDV, FBOD

IX. Conclusion

This is a very light weight protocol with minimum computational overheads. In DSDV, we need to maintain a routing table. AODV has a lot of overhead while discovering routes, which clogs the network for sending data packets to desired destination. Not only does no such complicacy exist in our protocol, but it also has some of their benefits. Like AODV it is an on-demand routing protocol and the physical hardware support needed to implement it is substantially low which increases its scalability. This protocol also has added features so as to nullify some of the security threats which cause faults in the MANET networks.

References

[1] [Perkins94] C.E. Perkins and P. Bhagwat, "Highly Dynamic Destination-Sequenced Distance-Vector Routing (DSDV) for Mobile Computers" , Comp. Comm. Rev., Oct.1994, pp.234-244.

[2] Luke Klein-Berndt, "A Quick Guide to AODV Routing"

[3] Rashid Hafeez Khokhar, Md Asri Ngadi and Satira Mandala, "A Review of Current Routing Attacks in Mobile Ad Hoc Networks", International Journal of Computer Science and Security, pp. 18-29, Volume-2 Issue-3

[4] "Avoiding Black Hole and Cooperative Black Hole Attacks in Wireless Ad hoc Networks" http://www.scribd.com/doc/26788447/Avoiding-Black-Hole-and-Cooperative-Black-Hole-Attacks-in-Wireless-Ad-hoc-Networks .

[5] Komala CR, Srinivas Shetty, Padmashree S., Elevarasi E., "Wireless Ad hoc Mobile Networks", National Conference on Computing Communication and Technology, pp. 168-174, 2010

[6] C. Perkins, E. Belding-Royer and S. Das, "Ad hoc On-Demand Distance Vector (AODV) Routing", 2003.

[7] Sapna S. Kaushik & P. R. Deshmukh. "Comparison of effectiveness of AODV, DSDV and DSR routing protocols is mobile Ad hoc networks", International Journal of Information Technology and Knowledge Management, July – December 2009, volume 2, No. 2, pp. 499-502.

[8] V. Ramesh, Dr. P. Subbaiah, N. Koteswar Rao, M. Janardhana Raju, "Performance Comparison and Analysis of DSDV and AODV for MANET", V. Ramesh et al. / (IJCSE) International Journal on Computer Science and Engineering, Vol. 02, No. 02, 2010, 183-188.

[9] Nor Surayati Mohamad Usop, Azizol Abdullah, Ahmad Faisal Amri Abidin, "Performance Evaluation of AODV, DSDV & DSR Routing Protocol in Grid Environment", IJCSNS International Journal of Computer Science and Netwprk Security, Vol. 9, No. 7, July 2009.

[10] Anuj K. Gupta, Dr. Harsh Sadawarti, Dr. Anil K. Verma, "Performance Analysius of AODV, DSR & TORA Routing Protocols", IACSIT International Journal of Engineering & Technology, Vol. 2, No. 2, April 2010, ISSN: 1793 - 8236.

[11] H. Yang, J. Shu, X. Meng, and S. Lu, "SCAN: Self-organized network-layer security in mobile ad hoc networks," IEEE Journal on Selected Areas in Communications, vol. 24, issue 2, pp. 261-273, February 2006.

[12] R. Balakrishan, S. Jayabalan, Dr. U. Rajeswar Rao, Dr. T. K. Basak. Dr. V. Cyrilraj, "Performance Issues on AODV and DSDV for MNAETS", Jounnal Theoritical and Applied Information Technoilogy.

[13] Angel R. Otero, Carlos E. Otero and Abrar Qureshi, "A Multi-Criteria Evaluation of Information Security Controls Using Boolean Features", International Journal of etwprk Security & its application (IJNSA), Vol. 2, No. 4, October 2010.

[14] Anand Patwardhan, Jim Parker, Michaela Iorga. Anupam Joshi, "Tom Karygiannis, Secure Routing and Intrusion Detection in Ad Hoc Networks" 3rd International Conference on Perv asive Computing and Communications (PerCom 2005), Kauai Island, Hawaii.

[15] Bing Wua, Jie Wua, Eduardo B. Fernandeza, Mohammad Ilyasa, Spyros Magliveras, "Secure and efficient key management in mobile ad hoc networks" Journal of Network and Computer Applications 30 (2007) 937–954.

[16] C. E. Perkins, E. M. Royer, and S. R. Das, "Ad Hoc On Demand Distance Vector (AODV) Routing," IETF Mobile Ad Hoc Networks Working Group, Internet Draft, work in progress, 17 February 2003.

[17] F. Anjum, Anup K. Ghosh, Nada Golmie, Paul Kolodzy, Radha Poovendran, Rajeev Shorey, D. Lee, J-Sac, "Security in Wireless Ad hoc Networks", IEEE journal on selected areas in communications, vol. 24, no. 2, February 2006.

[18] H. A. Wen, C. L. Lin, and T. Hwang, "Provably Secure Authenticated Key Exchange Protocols for Low Power Computing Clients," Computers and Security, vol. 25, pp. 106-113, 2006.

[19] Haiyun Luo, Petros Zerfos, Jiejun Kong, Songwu Lu, Lixia Zhang, "Self-securing Ad Hoc Wireless Networks", 7th IEEE Symp. on Comp. and Communications (ISCC), Taormina, 2002.

[20] Hongmei Deng, Wei Li, and Dharma P. Agrawal, "Routing Security in Wireless Ad Hoc Networks", IEEE Communications Magazine October 2002.

[21] Huaizhi Li Zhenliu Chen Xiangyang Qin, "Secure Routing in Wired Networks and Wireless Ad Hoc Networks" IEEE, 2004.

[22] Huaizhi Li, Mukesh Singha, "Trust Management in Distributed Systems" IEEE Computer Society February 2007.

[23] J. Nam, S. Cho, S. Kim, and D. Won, "Simple and Efficient Group Key Agreement Based on Factoring" Proc. Int'l Conf. Computational Science and Its Applications (ICCSA '04), pp. 645-654, 2004.

[24] Oscar F. Gonzalez, Michael Howarth, and George Pavlou, Detection of Packet Forwarding Misbehavior in Mobile Ad-Hoc Networks. Center for Communications Systems Research, University of Surrey, Guildford, UK. Integrated Network Management, 2007. IM '07. 10th IFIP/IEEE International Symposium on May 21, 2007.

[25] J. Parker, J. L. Undercoffer, J. Pinkston, and A. Joshi., "On Intrusion Detection in Mobile Ad Hoc Network s". In 23rd IEEE International Performance Computing and Communications Conference Workshop on Information Ass urance. IEEE, April 2004.

[26] Jeremy J. Blum, Member, IEEE, and Azim Es kandarian, Member, IEEE, "A Reliable Link-Layer Protocol for Robust and Scalable Intervehicle Communications" IEEE Transactions On Intelligent Transportation Systems, vol. 8, no. 1, March 2007.

[27] Jung-San Lee, Chin-Chen Chang, "Secure communications for cluster-based ad hoc networks using node identities" Journal of Network and Computer Applications 22 October 2006 International Journal of Computer Science and Security, Volume (1): Issue (1) 67.

[28] Kejun Liu, Jing Deng, Member, IEEE, Pramod K. Varshney, Fellow, IEEE, and Kashyap Balakrishnan, Member, IEEE, "An Acknowledgment-Based Approach for the Detection of Routing Misbehavior in MANETs" IEEE Transaction on Mobile Computing, VOL. 6, NO. 5, May 2007.

[29] Nik os Komninos, Dimitris Vergados, Christos Douligeris, "Layered security design for mobile ad hoc networks" journal computers & security 25, 2006 , pp. 121 – 130.

[30] P. Papadimitratos and Z.J. Haas, "Secure Link State Routing for Mobile Ad Hoc Networks" Proc . IEEE Workshop on Security and Assurance in Ad Hoc Networks, IEEE Press, 2003, pp. 27–31.

[31] Panagiotis Papadimitratos , Zygmunt J. Haas , "Secure message transmission in mobile ad hoc networks, Ad Hoc Networks" IEEE 2003, 193–209 .

[32] S. Holeman, G. Manimaran, J. Dav, and A. Chakrabarti, "Differentially secure multicasting and its implementation methods", Computers & Security, Vol 21, No. 8, pp 736-749, 2002.

[33] S. Matri, T. J. Giuli, K. Lai and M. Baker , Mittigating Routing misbehaviour in Mobile Ad Hoc Networks. *Proceedings of the 6th annual internaional conference on Mobile Computing and Networking (MOBICOM), Boston, Massachusetts, United States, 2000, 255-265.*

[34] Himadri Nath Saha, Dr Debika Bhattacharyya, Dr.P.K.Banerjee,"A Priority Based Protocol for Mitigating Different Attacks in MANET",International Journal for Computer Science and Communication,Volume I,Number2,pp-299-302,Sept.2010

[35] Himadri Nath Saha, Dr Debika Bhattacharyya, Dr.P.K.Banerjee,"A Distributed Administration Based Approach for Detecting and Preventing Attacks in MANET",International Journal for Scientific and Engineering Reasearch,Volume-2,Issue-3,pp-1-11,Mar-2011

[36] Himadri Nath Saha, Dr Debika Bhattacharyya, Dr.P.K.Banerjee,"Semi-Centralized Multi-Authenticated RSSI Based `Solution to Sybil Attack",International Journal of Computer Science and Emerging Technologies",Volume I,Issue-4,pp-338-341,Dec 2010.