A Secure Routing Protocol Based On Fidelity

Article

5 authors, including:

Abhik Jana
Indian Institute of Technology Kharagpur
9 PUBLICATIONS **4** CITATIONS
SEE PROFILE

Himadri Nath Saha
Institute of Engineering & Management
80 PUBLICATIONS **317** CITATIONS
SEE PROFILE

A Secure Routing Protocol Based On Fidelity

Authors: Mainak Biswas, Abhik Jana, Rajib Paul
Computer Science Department
Institute of Engineering and Management
Kolkata, India
E-Mail: *mnkbiswas@gmail.com, raajib_paul@yahoo.com*

Author: Himadri Nath Saha
Computer Science Department
Institute of Engineering and Management
Kolkata, India
E-Mail: *him_shree_2004@yahoo.com*

***Abstract*— Currently the mobile wireless technology is experiencing rapid growth. However the major challenge for deployment of this technology with its special characteristics is securing the existing and future vulnerabilities. The lack of static infrastructure causes several issues in mobile ad hoc network (MANET) environment, such as node authentication and secure routing. In this paper we propose a new protocol for secure routing of data packets in MANET. The approach is very simple, which will reduce the computational overhead to a lot extent. The heart of the protocol is a specific criterion of the nodes called "fidelity". We first explain what fidelity is and give a comprehensively detailed description of the protocol. Then exemplify the protocol with several case studies. And lastly discuss the security strengthening aspects of this simple yet robust protocol under some scenarios.**

Keywords- fidelity; sequence number; hop destination; flooding attack; black hole attack; co-operative black hole attack

I. INTRODUCTION

Mobile Ad-hoc Network (MANET) is a collection of wireless mobile hosts without fixed network infrastructure and centralized administration (Figure-1). Communication in MANET is done via multi-hop paths. Lots of challenges are there in this area: MANET contains diverse resources; the line of defense is very ambiguous; Nodes operate in shared wireless medium; Network topology changes unpredictably and very dynamically; Radio link reliability is an issue; 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 acts a router that forwards data packets to other nodes. Therefore selection of effective, suitable, adaptive and robust routing protocol is of utmost importance.

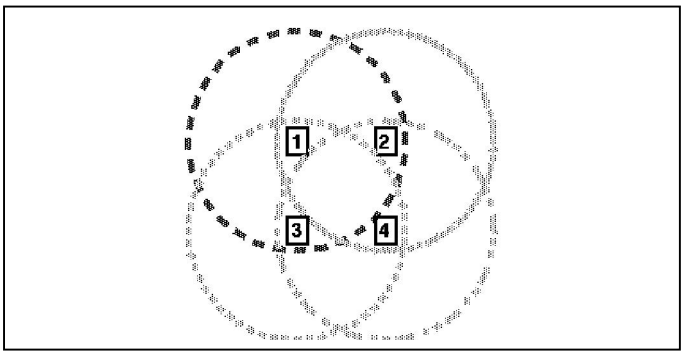

Figure 1: An ad-hoc mobile network with four nodes.

We now inspect what we mean by fidelity and how it helps to implement a routing protocol.

II. 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 to some extent. How fidelity looks after the security aspect, we will see that in the security aspects part of our paper. But for time being we will see what actually is fidelity and what does it denote.

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 occlude that the node density is also higher in that region for it is impossible to have one node having very high fidelity surrounded by nodes with low fidelity because a high fidelity 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.

III. DESCRIPTION OF THE PROTOCOL

To begin with, readers must understand that the term "friends of a node" used in this paper are actually the nodes that fall in the physical range of a particular node. When nodes are idle, after a very small stipulated time all the nodes will probe to understand which nodes are in its neighbourhood 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 of the nodes which are within its range and other is the fidelity value of that particular node. When each node has done that then they will sort that table according to the decreasing order of the fidelity value. Before we enter the detailed discussion of our protocol there are some concepts that need to be understood. They 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 of last n no. of received messages. After getting a message node will verify the tuple (source, sequence no) of that message with those tuples in its buffer. If anyone of them matches with that of the message then 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 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.

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.

IV. ALGORITHMS

Update friend list

- Send broadcast request for friends to reply
- Receive replies from neighbours
- Update my friend list
- Sort friend list

Generated data

- Set my status=busy
- 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
- Set my status=free

Received data

- If my status=busy send FAIL to sender
- Else
    - Make my status=busy
    - Process received data
    - Make my status=free

Process received data

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

- 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.
- 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.

V. Simulation Result

Simulated with JAVA in netbeans platform. Consider the following network topology as in figure-2. We show what happens when node 0 sends a message "hello" to node 3 in this network. In figure-2 the numbers written in the vicinity of nodes are node numbers, and the numbers written on the nodes are their corresponding fidelity values.

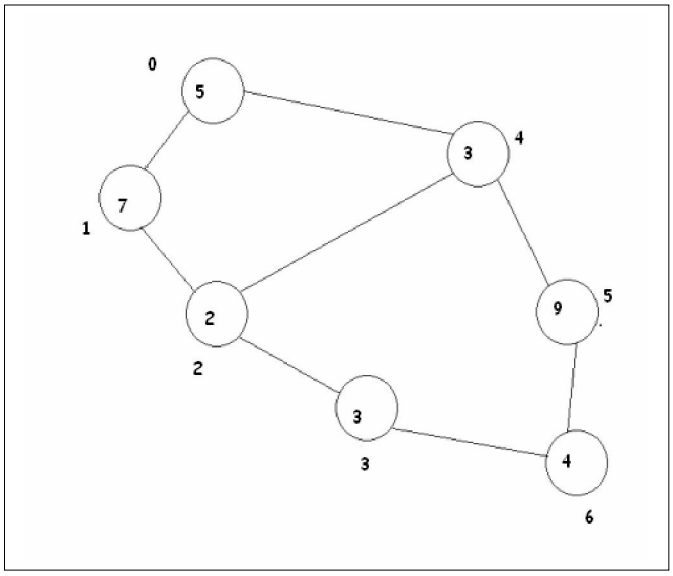

Figure 2- Network Topology (fidelity values written on nodes)

The events are:

- Generated at node:0 message: hello source:0 destination:3
- Destination:3 directly NOT reachable from node:0 friends sorted in descending order of reliability at node:0 are 1(7) 4(3)
- Sending from node:0 to node:1 message: hello source:0 destination:3
- Received at:1 from:0 message: hello source:0 destination:3
- Destination:3 directly NOT reachable from node:1 friends sorted in descending order of reliability at node:1 are 0(5) 2(2)
- Sending from node:1 to node:0 message: hello source:0 destination:3
- message: hello source:0 destination:3 discarded by node:0 as busy, sending FAIL
- Received FAIL at node:1 RESENDING
- Sending from node:1 to node:2 message: hello source:0 destination:3
- Received at:2 from:1 message: hello source:0 destination:3
- Destination:3 directly reachable from node:2 sending from node:2 to node:3 message: hello source:0 destination:3
- Received at:3 from:2 message: hello source:0 destination:3
- Accepted at node:3 message: hello source:0 destination:3
- Sending ACK from node:3 to node:2
- Received ACK at node:2
- Node:2 successfully forwarded message; new reliability=3
- Sending ACK from node:2 to node:1
- Received ACK at node:1
- Node:1 successfully forwarded message; new reliability=8
- Sending ACK from node: 1 to node: 0
- Received ACK at node: 0

VI. Security Aspects

A. *Flooding Attack*

Flooding attack is an attack that attempts to cause a failure in a computer system or other data processing entity by providing more input than the entity can process properly. Flood attacks occur when a network or service becomes so weighed down with duplicate packets that it can no longer process genuine connection requests.

In our protocol, we have two fields in the data packet *a)* Sequence number. *b)* Source node number. These two fields together uniquely identifies a message packet. Each data packet generated by a single node in this protocol is guaranteed to have different sequence number. Here if a node encounters a data packet that was recently 'seen' by it, it

discards the packet. This prevents propagation of duplicate packet through the network, mitigating flooding attack

B. Black Hole Attack

Black holes refer to places or nodes in the network where incoming traffic is silently discarded (or "dropped"), without informing the source that the data did not reach its intended recipient. The black hole problem is one of the security attacks that occur in MANET.

There are two possible solutions. The first is to exploit the packet sequence number included in any packet header. Computer simulation shows that compared to the original *ad hoc on-demand distance vector* (AODV) routing scheme. This solution can verify the route to the destination depending on the pause times at a minimum cost of the delay in the networks. The second is to find more than one route to the destination. Here we are going to discuss about how our protocol implements the second solution vividly.

Let us assume that there are four nodes A, B, C and D. A wants to send a data packet to node C. Node A cannot communicate with destination node directly. Data packet can be sent from A to C in two ways –

i) A→D→C
ii) A→B→C.

A will pick first path to send the data packet as fidelity of D is greater than B. A will wait for acknowledgement after sending the data packet to node D. But node D is a black hole node. Data packet can't be reached to destination node by this path.

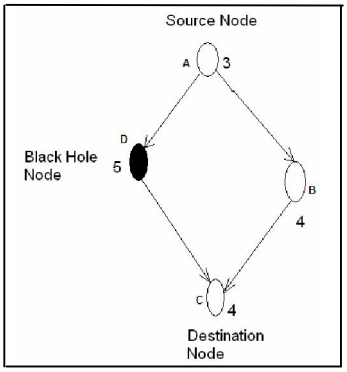

Figure no. 3.1

Data packet cannot be reached to the destination node by this path. Node A will pick the second path after timeout period shown in *figure no. 3.1*. Node A will send the data packet to node B and node B will relay the data packet to the destination node C. Node C will send an acknowledgement to node B. Node B will forward the acknowledgement to the source node A. After the successful transmission of data packet, the fidelity of node will be increased and updated. Thus, the fidelity of node B will increase to 5 from 4 as shown in *figure no. 3.2.*

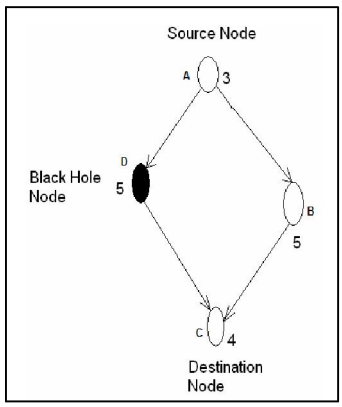

Figure no. 3.2

After every successful transmission of data packets the reliability and the fidelity of friend odes will increase. This way fidelity of node B will become 6 shown in *figure no. 3.3* after third transaction.

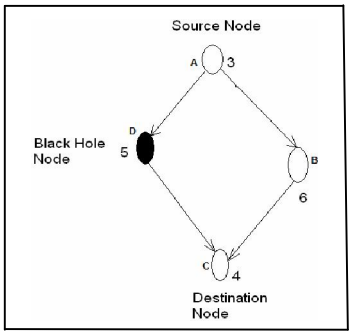

Figure no. 3.3

Afterwards node A will always pick node B to send data packet to node C as the fidelity of B is higher than node D. This way the black hole node will be terminated as shown in *figure no. 3.4.*

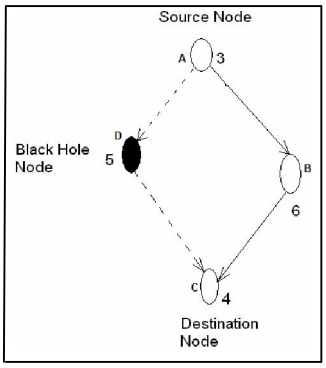

Figure no. 3.4

C. Co-operative Black Hole Attack

In co-operative black hole attack, there remains a group of malicious nodes. One of them takes the data packet and keeps on forwarding it among themselves so that the TTL of that

data packet finishes off and the data packet is automatically dropped. We understand that the modus operandi of these co-operative black hole attackers is to form a closed loop among themselves and keep on forwarding the message within this loop..

In our protocol a node will remain busy after sending a packet until it gets the acknowledgement for this message. So whenever any node try to send the message to a node already visited, it will simply reject the message and send a fail message to the sender as it is busy. As this protocol do not allow messages to transfer through the same node more than once there is no possibility for these malicious nodes to form the loop and hence co-operative black hole attack is mitigated.

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] 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.

[12] 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.

[13] 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

[14] 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

[15] 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.

[16] 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.

[17] 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.

[18] 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.

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

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

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

[22] 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.

[23] 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.

[24] 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.

[25] 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

[26] 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

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

[28] 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.

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

[30] 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.