# An Novel Approach to Determine Misbehaving Nodes in MANETs

**Karthik Jilla[1], Alefiah Mubeen[2]**

[1]Assistant Professor, Department of Computer Science & Engineering,
Kommuri Pratap Reddy Institute of Technology Rangareddy, Ghatkesar, Telangana, India.

[2]Assistant Professor, Department of Information and Technology,
Muffakhamjah College of Engineering & Technology, Hyderabad, Telangana State, India.

*Abstract---**In general, routing protocols for MANETs are designed based on the assumption that all participating nodes are two-way. While, due to the open structure and scarcely available battery-based energy, node misbehaviors may be present. Such routing misbehavior selfish nodes will participate in the route discovery but refuse to forward data packets. In this paper, we suggest the Two-ACK scheme that serves as an add-on procedure for routing schemes to discover misbehavior in selfish nodes and to moderate their adverse effect. The main idea of the Two-ACK scheme is to send 2-hop acknowledgment packets in the opposite direction of the routing path. Hence in order to reduce additional routing overhead, the received data packets are acknowledged in the Two-ACK scheme. The Two-ACK scheme identifiesmischievous nodes, and then seeks to recover the problem by notifying the routing protocol in future routes. We have found that, the Two-ACK scheme sensibly improves the packet delivery ratio, with some additional routing overhead.***

***Keywords: MANET, Two-Ack, Network, misbehavior, 2-hop, routing***

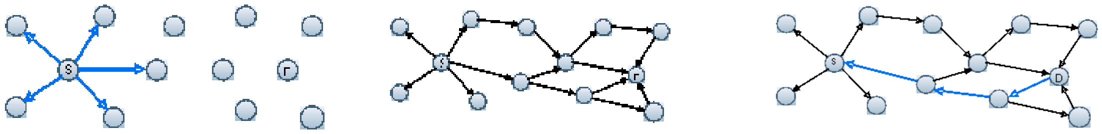
## I. INTRODUCTION

MANET is a group of mobile nodes, which communicate each other via wireless links either directly or depend on other nodes as routers. MANETs does not depend on pre-existing structure or base station nodes. Nodes in MANETs are free to move arbitrarily. Hence, the network topology of a MANET may change rapidly and randomly. All network activities, such as discovering and delivering data packets, are to be submitted by the nodes themselves, either independently or as a group. Depending on its application, the structure of network may vary from a small, to a large-scale, i.e..static network that is highly power-constrained mobile, highly dynamic network.

### *A. Characteristics of Manets*

Mobile ad-hoc networks are autonomous system connected by a wireless links, which forms a communication network. A MANET can be either a standalone entity or it can be an extension of wired network. There are many application areas of MANETs, such as:

1) *Military tactical operations* - For fast and possibly short term establishment of military communications for troop deployments in hostile and/or unknown environments.
2) *Search and rescue missions* - For communication in areas with little or no none.
3) *Disaster relief operations*- For communication in environments where the existing infrastructure is destroyed or left untreatable.
4) *Commercial use* - For communication purpose i.e.., in exhibitions, conferences, and at large gatherings.



(a) (b) (c)

Fig.1 Route Discovery

## II. PROBLEM OF ROUTING MISBEHAVIOR

The security problem and the misbehavior problem of wireless networks including MANETs have been studied by many researchers, e.g., ([10], [11]- [14], [16]). Various techniques have been proposed to prevent selfishness in MANETs.

### *A. Routing Misbehavior Model*

Selfish node does not perform the packet forwarding function for the data packets unrelated to it. However, it operates normally in the Route Discovery and the Route Maintenance phases of the DSR protocol. Since such nodes participate in the Route Discovery phase, they may be included in the routes chosen to forward the data packets from the source to destination. The selfish nodes, however, refuse to forward the data packets from the source. This leads to ambiguity in source node.

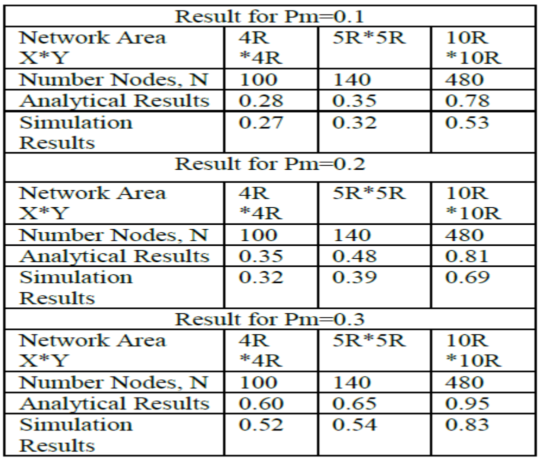
The attackers (misbehaving nodes) are assumed to be capable of performing the following

1) Dropping any data packet.
2) Sending out fabricated Two-ACK packets.
3) Sending out fabricated hn, the key generated by the Two-ACK packet senders.
4) Claiming falsely that its neighbor or next-hop links are misbehaving tasks.

## III. IDENTIFYING MISBEHAVING LINKS USING TWO-ACK SCHEME

The ACK packets at the TCP layer have a similar effect as our 2-ACK packets do. The main differences are the following: First, ACK packets in TCP are used for the purpose of flow-control and reliable end-to-end communication, at the same time as selfishness is big problem that should be solved by the underlying IP layer. In the absence of a lower layer acknowledgment scheme, the source and other intermediate nodes have no way of finding out which of the downstream nodes is misbehaving. It will be inefficient to conclude that the entire route is misbehaving when indeed there is only one misbehaving node. To correctly detect and isolate such a misbehaving node, additional techniques such as the Two-ACK scheme need to be employed. Second, ACK packets in TCP have to travel all the way from the final destination back to the source. Therefore, depending on the length of the path used for data packets, it is likely that ACK packets will arrive after significant delays. In contrast, Two-ACK packets travel exactly two hops, making the timeout period shorter and more predictable. To detect misbehavior, the sender or router of a data packet maintains a list of data packet IDs that have yet to receive a Two-ACK acknowledgment packet from a node two hops away. Each node maintains a unique list for each forwarding link that it is using. Each item on the list has the following data members shown in Table 1.

TABLE 1: FORWARDING LINKS



| Result for Pm=0.1 | | | |
|---|---|---|---|
| Network Area X*Y | 4R *4R | 5R*5R | 10R *10R |
| Number Nodes, N | 100 | 140 | 480 |
| Analytical Results | 0.28 | 0.35 | 0.78 |
| Simulation Results | 0.27 | 0.32 | 0.53 |
| Result for Pm=0.2 | | | |
| Network Area X*Y | 4R *4R | 5R*5R | 10R *10R |
| Number Nodes, N | 100 | 140 | 480 |
| Analytical Results | 0.35 | 0.48 | 0.81 |
| Simulation Results | 0.32 | 0.39 | 0.69 |
| Result for Pm=0.3 | | | |
| Network Area X*Y | 4R *4R | 5R*5R | 10R *10R |
| Number Nodes, N | 100 | 140 | 480 |
| Analytical Results | 0.60 | 0.65 | 0.95 |
| Simulation Results | 0.52 | 0.54 | 0.83 |

### *A. Probability of Misbehaving Routes*

In order to determine the effect of routing misbehavior, we estimate the probability of misbehaving routes in this subsection. A route is denied as misbehaving when there is at least one router along the route that can be classified as selfish. Our analysis is based on the few following assumptions:

1) The network nodes are randomly distributed over the entire network. Each node's location is independent of all other nodes' locations. There are N nodes in the network area of size X * Y
2) The source and the destination of each transaction are chosen randomly among all nodes;
3) Other than source and destination node, a new node is selected as misbehaving node, independently, with probability pm.

We examine a route with an average number of hops, h. There are h -1 router between the source and the destination. Each of these routers may misbehave with probability pm. The probability of the route with at least one misbehaving node is:

$$P_r = 1 - (1 \square p_m)^{h-1}$$

### *B. Modules of Two-ACK algorithm*

The modules are

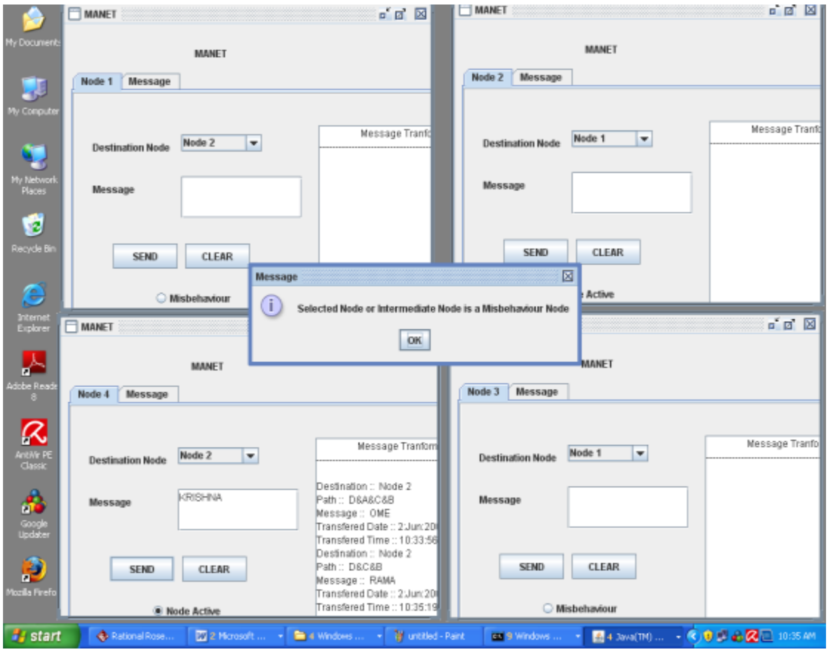
1) Route Request for identifying the misbehavior Node,
2) Message Transfer to the available path, Route Maintenance,
3) Cache Updating for misbehavior node identification

Route Request for identifying the misbehavior Node: When a source node wants to send packets to a destination, it initiates a Route Discovery by broadcasting a ROUTE REQUEST, to send messages which it does not have a route. The node receiving a ROUTE REQUEST validate whether it has a route to the destination in its cache and also check is it misbehavior node or not. If it has, it sends a ROUTE REPLY to the source including a source route. If the node does not have a cached route to the destination, then it adds its address to the source route and rebroadcasts the ROUTE REQUEST. When the destination receives the ROUTE REQUEST, it sends a ROUTE REPLY with the source route . Each node forwarding a ROUTE REPLY stores the route from starting node to destination node. When the source receives the ROUTE REPLY, it caches the source route. If any node not sends acknowledgement then we easily identified that is misbehavior node. So find out the alternative path and forwarding the data to the destination.

*A. Message Transfer to the available path*: The Message transfer relates with that the sender node wants to send a message to the destination node after the path is selected also find out that node is not a misbehavior node and status of the destination node through is true. The receiver node receives the message completely and it sends the acknowledgement to the sender node and also to nearby nodes through the router nodes where it is received the message.

*B. Route Maintenance:* Route Maintenance, the node forwarding a packet is responsible for confirming that the message (packet) has been received successfully by the next hop. If no acknowledgement is received after maximum retransmissions, then the forwarding node sends a ROUTE ERROR to the source node, indicating that the link is broken. Each node forwarding the ROUTE ERROR removes from its cache the routes containing the broken link.

Cache Updating for misbehavior node identification: When a node detects a link failure, our goal is to alert all reachable nodes that have cached that link, to update their caches. To achieve this goal, the node detecting a link failure, identifying the selfish node needs to know which nodes have cached the broken link and needs to notify such nodes efficiently. Our solution is to keep track of identifying the misbehavior node in a distributed manner. The results are shown in Figure 4 shows the Message sending from Node3 to Node2 via the PATH 'A & C & B'. In this path A behaves like Misbehave Node then it fine another path, the other path contains misbehave node then display this message



Message

Selected Node or Intermediate Node is a Misbehaviour Node

Destination Node: Node 2
Message: KRISHNA

Destination :: Node 2
Path :: D&A&C&B
Message :: OME
Transfered Date :: 2:Jun:20
Transfered Time :: 10:33:56
Destination :: Node 2
Path :: D&C&B
Message :: RAMA
Transfered Date :: 2:Jun:20
Transfered Time :: 10:35:19

Fig. 2 Actual results of the process

## IV. CONCLUSION & FUTURE WORK

In this paper, we have investigated the performance degradation caused by such selfish (misbehaving) nodes in MANETs. We have proposed and evaluated a technique, termed Two-ACK, to detect and mitigate the effect of such routing misbehavior. The Two-ACK technique is based on a simple 2-hop acknowledgment packet that is sent back by the receiver of the next-hop link. The 2-ACK scheme can be used as an add-on technique to routing protocols such as DSR in MANETs. In this work, we focused only on link misbehavior. It is more difficult to decide the behavior of a single node. This is mainly due to the fact that communication takes place between two nodes, and is not the only effort of a single node. Hence, utmost care must be taken before punishing any node associated with the misbehaving links. In order to decide the behavior of a node and punish it, In future we may need to check the behavior of links around that node.

## REFERENCES

[1] H. Miranda and L. Rodrigues, "Preventing Selfishness in Open Mobile Ad Hoc Networks," *Proc. Seventh CaberNet Radicals Workshop*, Oct. 2002.

[2] S. Marti, T. Giuli, K. Lai, and M. Baker, "Mitigating Routing Misbehavior in Mobile Ad Hoc Networks," *Proc. MobiCom*, Aug. 2000.

[3] D. Johnson, D. Maltz, Y.C. Hu, and J. Jetcheva, "The Dynamic Source Routing Protocol for Mobile Ad Hoc Networks," *Internet draft*, Feb. 2002.

[4] L. Zhou and Z.J. Haas, "Securing Ad Hoc Networks," *IEEE Network Magazine*, vol. 13, no. 6, Nov./Dec. 1999.

[5] F. Stajano and R. Anderson, "The Resurrecting Duckling: Security Issues in Ad-Hoc Wireless Networks," *Proc. Seventh International Workshop Security Protocols*, 1999.

[6] J. Kong, P. Zerfos, H. Luo, S. Lu, and L. Zhang, "Providing Robust and Ubiquitous Security Support for Mobile Ad-Hoc Networks," *Proc. IEEE Internal Conference Network Protocols (ICNP '01)*, 2001.

[7] Aad, J.-P. Hubaux, and E-W. Knightly, "Denial of Service Resilience in Ad Hoc Networks," *Proc. MobiCom*, 2004.

[8] L. Buttyan and J.-P. Hubaux, "Stimulating Cooperation in Self-Organizing Mobile Ad Hoc Networks," *ACM/Kluwer Mobile Networks and Applications*, vol. 8, no. 5, 2003.

[9] B. Awerbuch, D. Holmer, C.-N. Rotaru, and H. Rubens, "An On- Demand Secure Routing Protocol Resilient to Byzantine Failures," *Proc. ACM Workshop Wireless Security (WiSe)*, Sept. 2002.

[10] S. Marti, T. Giuli, K. Lai, and M. Baker, "Mitigating Routing Misbehavior in Mobile Ad Hoc Networks," *Proc. MobiCom*, Aug. 2000.

[11] S. Marti, T. Giuli, K. Lai, and M. Baker, "Mitigating Routing Misbehavior in Mobile Ad Hoc Networks," *Proc. MobiCom*, Aug. 2000.

[12] L. Buttyan and J.-P. Hubaux, "Enforcing Service Availability in Mobile Ad-Hoc WANs," *Proc. MobiHoc*, Aug. 2000.

[13] J.-P. Hubaux, T. Gross, J.-Y. LeBoudec, and M. Vetterli, "Toward Self-Organized Mobile Ad Hoc Networks: The Terminodes Project*," IEEE Comm. Magazine*, Jan. 2001.

[14] S. Buchegger and J.-Y. Le Boudec, "Performance Analysis of the CONFIDANT Protocol: Cooperation of Nodes, Fairness in Dynamic Ad-Hoc Networks," *Proc. MobiHoc*, June 2002.

[15] S. Zhong, J. Chen, and Y.R. Yang, "Sprite: A Simple, Cheat-Proof, Credit-Based System for Mobile Ad-Hoc Networks," *Proc. INFOCOM*, Mar-Apr. 2003.

[16] M. Jakobsson, J.-P. Hubaux, and L. Buttyan, "A Micropayment Scheme Encouraging Collaboration in Multi-Hop Cellular Networks," *Proc. Financial Cryptography Conf*erence, Jan. 2003.

KARTHIK JILLA Received the B.Tech with first class in CSE from CJITS, Jangaon, in 2008 and Masters in Computer Networks with first class from Middlesex University, London, U.K. in 2011 and M.Tech with Distinction in Software Engineering from VBIT, Jangaon, in 2013. He has Teaching experience of 5 years and currently working as Assistant Professor in Kommuri Pratap Reddy Institute of Technology, Hyderabad. He is a lifetime member of International Society for Research and Development (ISRD), a Professional Society bearing membership Id: M4150902185.

ALEFIAH MUBEEN, received B. Tech degree with a silver medal in Information Technology from Vidya Jyothi Institute of Technology in 2007 and M Tech in Computer Science with two gold medals from University of Hyderabad in 2010. She worked as a Systems Engineer for 2 years in TCS, Hyderabad. Since October 2012 she is working as Assistant Professor in Muffakham Jah College of engineering and technology. She is a member of Computer Science Teachers Association (CSTA) member no: 611696. She is also a lifetime member of International Association of Computer Science and Information Technology (IACSIT) bearing member no: 80351031.