IOSR Journal of Computer Engineering (IOSR-JCE)
e-ISSN: 2278-0661, p- ISSN: 2278-8727Volume 9, Issue 1 (Jan. - Feb. 2013), PP 53-60
www.iosrjournals.org

Detection of routing misbehavior in MANET using improved 2ACK

Prof. Poonam Gupta[1], Sarita Chopde[2]
[1](Computer Engineering Department, GHRCEM College, Pune University , India)
[2](Computer Engineering Department, GHRCEM College, ,Pune University , India)

Abstract: *A mobile ad hoc network consists of individual mobile nodes communicating via wireless link. The node which refuses to share their own resources but gets benefit from other nodes, are called selfish or misbehaving nodes. The nodes in MANET are mobile with changing topologies. Performance of network may get affected due to these selfish nodes like scarcely available battery-based energy and node misbehaviors may exist. The 2ACK scheme is used to detect the misbehaving link. In this paper, improved 2ACK scheme detects misbehaving node.*

It sends two-hop acknowledgment packets in the opposite direction of the routing path. Improved 2ACK scheme reduces additional routing overhead by minimizing acknowledgment of the received data packets. It uses the Dynamic Source Routing (DSR) protocol. The proposed improved 2ACK scheme tries to reduce the overhead of Acknowledgments caused by 2ACK scheme.

Keywords *- Dynamic source routing, Mobile Ad Hoc Networks (MANET), network security, Node misbehavior, routing misbehavior*

I. INTRODUCTION

The In mobile wireless networks, it is easy for a node to enter into an ad hoc network which causes mesh confusion. As there is no dedicated authority for routing, packet forwarding, authentication and network management, security becomes an important issue. In network, node often changes their location which can leads to unnecessary routing overhead. For instance the Bluetooth or wireless network is free to use though many of organizations can't use it as their organizational network. In such cases 2ACK with DSR can be used to find misbehavior.

Here we are detecting the selfish node with the MANET architecture which represents host and routers connected by classic IP link in the given Ethernet. The hosts are connected either internally or externally while routers are connected via MANET interface. MANET interface connects routers. Classic IP connects Router to host. By getting same configuration and properties from previous host through classic IP link, each host act as basic host in other network. In MANET, the performance may degrade due to refusal from nodes to forwarding data ahead. To overcome this problem, DSR uses the watchdog and the path rater. Watchdog might not detect a misbehaving node in the presence of ambiguous collisions, receiver collisions, limited transmission power, false misbehavior, collusion, and partial dropping. To detect this DSR is used with 2ACK.

TYPES OF MANET

A. Closed MANET

In a closed MANET, all mobile nodes cooperate with each other towards a common goal, such as emergency search/rescue or military and law enforcement operation.

B. Open MANET

In an open MANET, different mobile nodes with different goals share their resources in order to ensure global connectivity. But in some cases the nodes refuses to share its data, called as selfish nodes or misbehaving nodes.

There are two categories: credit-based schemes and reputation-based schemes to avoid the selfish nodes.

1) Credit Based Scheme:

For performing network functions credit based scheme provides incentive for nodes, faithfully.

2) Reputation Based Scheme:

Here in this scheme network detects and declare the misbehaving nodes in collective manner and then these nodes are cut off from the network. [4][5]

Watchdog and path rater are the techniques to detect and mitigate, respectively, routing misbehavior in MANETs.

3) End-to-End Acknowledgment Schemes

It contains, the acknowledgments (ACKs): to detect routing misbehavior and the Selective Acknowledgment (SACK): technique to acknowledge out-of- order data blocks.

4) The TWOACK and S-TWOACK Schemes

It contains, TWOACK: TWOACK packets are sent for every data packet received, S-TWOACK: each TWOACK packet in the S-TWOACK scheme acknowledges the receipt of a number of data packets. [10]

II. DYNAMIC SOURCE ROUTING

The Dynamic Source Routing protocol is designed for use in multi-hop wireless ad hoc networks of mobile nodes. [3]DSR helps nodes in self organizing and self configuring. It works in two steps.

A. *Route Discovery:*

Route discovery is done only when packet is being sent from source to destination where the packet doesn't know the path already.

B. *Route Maintenance:*

Here the route between source and destination is maintained by checking the network topologies, if route is broken source can make attempt to find another route for destination.

Each of Route Discovery and Route Maintenance operate entirely on demand.

III. 2-ACK SCHEME

The 2 ACK (acknowledgement) is the latest version of all ACKs schemes. It detects the misbehavior routing and confidentiality of data message in MANET environment. The proposed work (2ACK with confidentiality) is as follows.

• Using 2ACK if the time required for sending data is less than the wait time and intermediate node contains same message as original, the sender is massaged that the link is working properly.

• Using 2ACK if the time required for sending data is more than the wait time and intermediate node contains same message as original, the sender is massaged that the link is misbehaving.

• Using 2ACK if the time required for sending data is more than the wait time and intermediate node doesn't contain same message as original, the sender is massaged that the link is misbehaving and confidentiality is lost.

• Using 2ACK if the time required for sending data is less
than wait time and intermediate node doesn't contain same message as original, the sender is working properly and confidentiality is lost.

At destination, a hash code will be generated which will be compared with the sender's hash code for checking the confidentiality of message. Therefore if link is misbehaving, sender will not use it in future and hence loss of packets can be avoided.

A. System Model

While sending data from source to destination via intermediate link node, it may possible that the selected link is encountering some problems like,

- Data packet is not forwarded from intermediate to destination
- Time can be consumed more than expected
- Data may modify during transmission.

In MANET, there is no retransmission hence care should be taken that data must not lose. And due to misbehaving of either sender or receiver, the next hop may have adverse effect on its working that can cause link tagging.

Our approach is used to discuss the significantly simplification of the routing detection mechanism and also checking the confidentiality of the message in MANETs environment.[2]

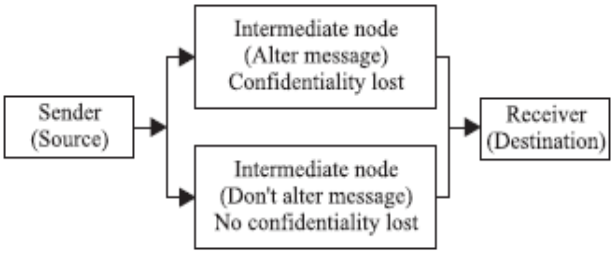

Fig1. System model

1) Module 1: Sender module (Source node)

- Reads message and divides it in 48 bytes
- Sends it to receiver via intermediate node and receives ack.

- "Cpkts" counter is incremented by 1 after each sending.
- If 2ACK time<wait time. "Cmis" counter is incremented by 1.
- Rmis is threshold ratio
- If Cmis/Cpkts < Rmis, the link is working properly otherwise it is misbehaving.

2) *Module 2: Intermediate module (Intermediate node).*
- Receives packet from sender
- Alter/don't alter
- Receives 2ACK packet from receiver and sends 2 acknowledgements to sender.

3) *Module 3: Receiver module (Destination node).*
- Receives message from intermediate node
- Take out destination name and hash code
- Decoding of hash code of destination and compares it with source's
- Sends 2ACK to source via intermediate node. The task of this module is to receive message from the intermediate node, take out destination name and hash code and decode it. Compare the hash code of source node and destination node for security purpose. Send 2ACK to source through the intermediate node.

B. Working of 2ACK Scheme

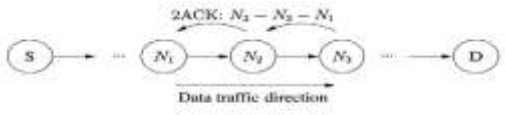

Fig2- 2ACK Scheme

Figure 3 illustrates the operation of the 2ACK scheme. Suppose that N1, N2 and N3 are three consecutive nodes (triplet) along a route. The route from a source node, S, to a destination node, D, is generated in the Route Discovery phase of the DSR protocol. When N1 sends a data packet to N2 and N2 forwards it to N3, it is unclear to N1 whether N3 receives the data packet successfully or not. Such an ambiguity exists even when there are no misbehaving nodes. The problem becomes much more severe in open MANETs with potential misbehaving nodes. [1]

On the successful reception of the data packet ID, N3 sends explicit acknowledgment to N1 for its notification in 2ACK scheme. Node N3 sends 2ACK packets in opposite direction to N1 after successful reception of data packets. Hence triplet [N1 → N2 → N3] is formed from the original data traffic route where N1, observing node checks the link N2 → N3. In 2ACK scheme, data transmitted through the triplet along the route where any node can be acts as a sender or receiver. [6]

For detecting misbehavior, the 2ACK packet sender having ID list of forwarded data packets but no record of acknowledged. It maintains Cpkts, counter which is incremented for forwarded data packets within timeouts i.e. τ seconds and Cmis counter which is incremented simultaneously when data packets are not forwarded within timeouts. According to that, nodes are added and removed from the ID list.

After receiving a data packet by N3, it takes decision for sending 2ACK packets to N1 and then it will be acknowledged only fraction of data packets via 2ACK packets. This fraction of data packet is called acknowledgment ratio, Rack. The overhead of 2ACK packet transmission is maintained by varying Rack.

By varying Rack, it is possible to tune dynamically the overhead of 2ACK packet transmissions. For a period of time termed Tobs, Node N1 observes the behavior of link N2 → N3. At the end of Tobs, N1 performs calculation of Cmis/Cpkts *i.e.* the ratio of missing 2ACK packets and compares it with a threshold Rmis. If Cmis/Cpkts >Rmis, then link N2 → N3 is declared misbehaving and N1 sends out an RERR packet.

Since only a fraction of the received data packets are acknowledged Rmis should satisfy Rmis > 1 – Rack in order to eliminate false alarms caused by such a partial acknowledgment technique.

Each node receiving or overhearing such an RERR marks the link N2 → N3 as misbehaving and adds it to the blacklist of such misbehaving links that it maintains. When a node starts its own data traffic later, it will avoid using such misbehaving links as a part of its route.

1) Authenticating the 2ACK Packets

When the 2ACK packets are forwarded by an intermediate node without proper protection, a misbehaving node *N*2 can simply fabricate 2ACK packets and claim that they were sent by node *N*3. Therefore, an authentication technique is given through the digital signature algorithm to protect 2ACK packets from being forged. A digital signature is a small number of extra bits of information attached by node *N*3. The signature is unique and usually computationally impossible to forge unless the security key of node *N*3 is disclosed. [1]

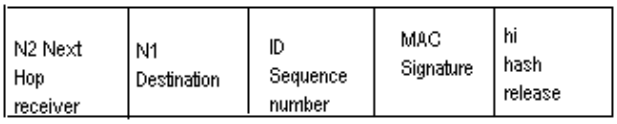

Fig3-ACK packet format

2) Acknowledgment Ratio, Rack

The additional routing overhead caused by the transmission of the 2ACK packets can be controlled by the parameter acknowledgment ratio, Rack, at the 2ACK packet sender. By using the parameter Rack in the 2ACK scheme, only a fraction of the received data packets will be acknowledged. Means Rack provides a mechanism to tune the overhead. The reduction of overhead comes with a cost: the shrinking of the range over which Rmis can take values.[1]

3) Partial Data Forwarding

Sometimes selfish node may cheat by forwarding wrong data. Now if N2 is misbehaving which is receiving data from N1, will forward fraction of data to N3 Rpart *(*0 < Rpart < 1). Thus, *N*3 receives Rpart, *D*ata packets and only Rack. Rpart data of them will be acknowledged by 2ACK packets sent from *N*3. Therefore, in order to cheat the system, a misbehaving node *N*2 has to make sure that as the gap between 1 – Rack and Rmis shrinks, the feasible value of Rpart approaches 1. Hence 2ACK guards the partial forwarding.

4) Timeout for 2ACK Reception, τ

The parameter timeout, *τ*, will be used to set up a timer for 2ACK reception. If packet is received before the expected 2ACK, the timer expires and the missing 2ACK packet counter, Cmis, will be incremented. Thus, τ value matters. If timeout is too small false alarms are triggered on contrary node will have to maintain longer list and memory.

Therefore, τ should be set at a value that is large enough to allow the occurrence of temporary link failures (for example, the unsuccessful transmission due to node mobility or local traffic congestion).

A single-hop transmission delay includes packet transmission delay, random back-off delay at the Medium Access Control (MAC) layer, data processing delay, and potential retransmission delay. Hence timeout value should be maintained.

5) *Observation Period*, Tobs, and Dynamic Behavior

In the 2ACK scheme Tobs distinguishes link misbehaviors and temporary link failures by observing the reception of 2ACK packets over a certain period of time. Such a technique is able to distinguish temporary link failures from link misbehavior.

IV. IMPROVED 2-ACK SCHEME

The Improved 2 ACK scheme uses the concept of 2ACK scheme as it is based on it. The improved 2 ACK scheme is used for detecting misbehaving link or node in triplet. [7] In 2 ACK scheme algorithm, two nodes has to keep track of acknowledgement. To reduce number of ACK and detecting which node is exactly misbehaved in triplet, we come towards improved 2 ACK scheme.

A. WORKING OF IMPROVED 2-ACK SCHEME

The improved 2ACK scheme is used to detect misbehaved node as well as reduce the number of acknowledgement. For this, there are 3 possibilities: The Improved 2 ACK scheme uses the concept of 2ACK scheme as it is based on it. The improved 2 ACK scheme is used for detecting misbehaving link or node in triplet. [7] In 2 ACK scheme algorithm, two nodes has to keep track of acknowledgement. To reduce number of ACK and detecting which node is exactly misbehaved in triplet, we come towards improved 2 ACK scheme

1) Best case:

In this case, let's assume that there is no misbehavior in triplet. Suppose time ™ is required to send packet and receive ACK between two consecutive nodes. The packet will be sent by N1 to N2 and will be forwarded by N2 to N3. Then N3 will send ACK in reverse path (i.e. N3->N2->N1) Here N2 will not send its own ACK to N1.

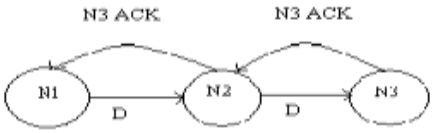

Fig.4 No misbehavior

2) *When N3 Misbehaved*: In the above case, let's consider N3 will misbehave(i.e. it will drop either packet or will not send ACK). In this case, N2 will wait for N3's ACK for time ™ and if it is not getting then N2 will send its own ACK to N1 which informs N1 that N3 is misbehaving as N1 is getting ACK of N2 and not of N3.

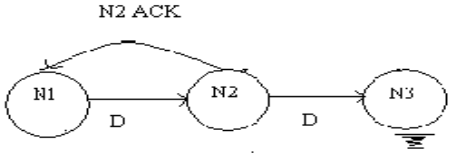

Fig.5. N3 misbehaving

3) Worst Case: Suppose N2 misbehaves (i.e. Either N2 drops the packet or it drops ACK sent by N3). In both cases N2 can't send ACK to N1 which will inform N1 after time 2TM (time starts from packet sent from N1 to *N2) N2 is misbehaving.*

Fig.6. N2 misbehaving

B. Advantages of Improved 2ACK Scheme

It solves the problems of ambiguous collisions, receiver collisions, and limited transmission power: Thus it is a kind of disadvantage which hampers the receiver collisions performance and disturbs other packets performance in acknowledging.

1) Limited Transmission Power-

A misbehaving *N*2 may maneuver its transmission power such that *N*1 can overhear its transmission but *N*3 cannot. It becomes a threat only when the distance between *N*1 and *N*2 is less than that between *N*2 and *N*3. The 2ACK scheme is immune to limited transmission power problem.

2) Limited Overhearing Range-

The I2ACK scheme is immune to the limited overhearing range issue caused due to low transmission power in the communication link.

3) Routing Overhead-

Disadvantage of the 2ACK scheme has a higher routing overhead which is reduced with the help of I2ACK.

4) Acknowledgement-

This additional routing overhead is caused by the transmission of 2ACK packets. So by reducing the acknowledgment ratio, Rack, the number of 2ACK transmissions can be significantly lowered overhead.

C. *Graph:* In the graph, we considered six nodes and observe the result. In best case, the graph shows that the total number of acknowledgments of improve 2 ACK gets reduced than other such as 2 ACK and NACK scheme. In the figure 7, we considered best case with delay acknowledgment which reduces ACK in initial stage.

In next figure, graph again plotted no. of nodes vs. total ACK received with delay acknowledgment. This graph displayed that Improved 2ACK is better than 2ACK scheme as it reduces acknowledgment.

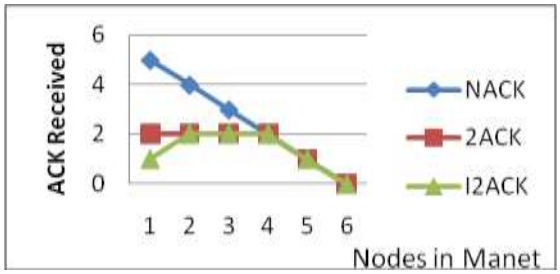

Fig.7 Best case with delay

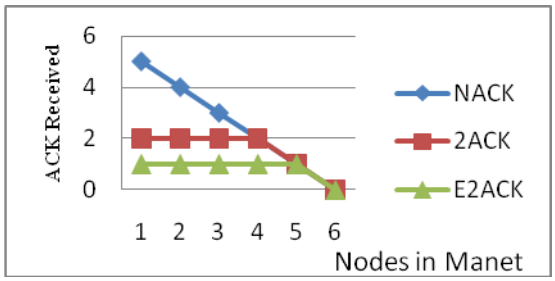

Fig. 8 ACK with No Delay

V. CONCLUSION AND FUTURE SCOPE

This Improved 2 ACK scheme is better than previous 2ACK scheme as it reduces the acknowledgement and also display exactly misbehaved node. This scheme also reduced the number of ACK when it acts in best case without delay as it sent acknowledgement within time.

There are various other routing schemes to implement the improved2ACK scheme. But here DSR are used as basic protocol for implementing 2ACK scheme. The main challenge is how to derive the information so that the 2ACK sender and the observing node are informed of such information.

In future work, there will be investigation of how to add the improved 2ACK scheme to other types of routing schemes and open networks. Also there will be investigation of new algorithm which reduces the overhead of intermediate node.

ACKNOWLEDGEMENTS

Apart from my own, the success of this report depends largely on the encouragement and guidelines of many others. I am especially grateful to my guide Prof. Poonam Gupta and Head of Computer Engineering Department, GHRCEM, who has provided guidance, expertise and encouragement. I express my heartfelt gratefulness to my guide, for her stimulating supervision whether required during my seminar work. I am also thankful to all the staff of Computer Engineering Department for their cooperation and support. I would like to put forward my heartfelt acknowledgement to all my classmates, friend and all those who have directly or indirectly provided their overwhelming support during my seminar work and the development of this report.

VI. PSEUDO CODE OF IMPROVED 2ACK SCHEME

The triplet N1 → N2 → N3 in Figure illustrate 2ACK's processing. Note that such codes are run on each of the sender/receiver of the 2ACK packets.

A. 2ACK Packet Sender Side (Node N3)

Step1: publish hn // Send authenticated element to N1
Step2: Cpkts = 0, Cack = 0, i= n , Initialization at node N3
Step 3: if (data packet received) then go to step 5.
Step 4: If(dest addr = self) Increase the counter Cpkts
Step 5: if (Cack = Cpkts < Rack) then the data packet acknowledged needs to be prepare MAC with hi–1
prepare 2ACK with ID, MAC, hi,
Add authentication to 2ACK packet
Step 6: send ack
make hop counters 0
stop triplet process
else
Cpkts ++ // Increase the counter of received packets
send ack
Step 7: if (Cack/Cpkts < Rack) // now acknowledge the data packets
Step 8: prepare MAC with hi-1
Step 9: prepare ACK // ACK creation
Step 10: send ACK
Step 11: Cack ++, i - - //push the counter maintained for
acknowledged packets
Step 12: end
Step 13: end
Step 14: end

B) *Receiver (Observer) Side (Node N1)*

Step 15: while true do
Step 16: if(packet in otput buffer)
Step 17:send packet to N2
wait for time 2tm+te(tm is time req to send packet
and recv ack to nearest node in path And te is extended time prosees time)
Step 18: if ACK received
Step 19: check for ACK, if ACK from N2
Step 20: display msg. N3 misbehaving and send misbehave info to source
Step 21: else if ack from N3
Step 22: No misbehavior
Step 23: release triplet and make new triplet from N2
Step 24: stop

C) Parallel process 1 (receiving hn)

Step 25: While (true) do.
Step 26: if receive hn from the ACK packet sender then
Record hn, i ← n

D) *Parallel process 2 (receiving 2ACK packets)*
while true do
step 27: randomly select T start > current time
Step28: Start the observation
Step 29: up to current time < T start do
Step 30: // null
Step31: Stop
Step 32: LIST will Initialization at node N1
Step33: while current time < T start + Tobs do // Observation period is not expired
Step 34: if (data packet forwarded) then
Step 35: LIST ← LIST U data ID // Add a data ID to LIST
Step36: Increase the counter of forwarded packets
Step 37: setup timer (τ) for data ID // Record the time
Step 38: stop
Step 39: if (timeout event happens) then // 2ACK packet for a data ID is not received
Step 40: Remove data ID from LIST
Step 41: // Increase misbehavior counter
Step 42: end
Step 43: end
Step 44: if (Cmis / Cpkts > Rmis) then the observation period expires
Step 45: send link misbehavior report
Step 46: Stop
Step 47: Stop

E*) For Node N2:*
Step 49: if data pkt received
Step 50: chk for dest, if(dest address=self)
Step 51: same as N3
Step 52: else copy packet and forward to N3
Step 53: wait for(tm+te), start timer
Step 54:if ack recieved within (tm+te)
Step 55:forward ack of N3 to N1
Step 56: else send self ack
Step 57: Stop
Step 58: Stop

REFERENCES

Journal Papers:

[1] K. Liu, J. Deng, P. K. Varshney, and K. Balakrishnan, "An Acknowledgment-based Approach for the Detection of Routing Misbehavior in MANETs," IEEE Transactions on Mobile Computing, vol. 6, no. 5, pp. 536-550, May 2007.Tavel, P. 2007 Modeling and Simulation Design. AK Peters Ltd.

[2] Sunilkumar , S. Manvia, Lokesh B. Bhajantrib, and Vittalkumar K. Vaggac, "Routing Misbehavior Detection in MANETs Using 2ACK", journal of telecommunication and information technology 4/2010

[3] David B., Johnson David, A. Maltz, Josh Broch, "DSR: The Dynamic Source Routing Protocol for Multi-Hop Wireless Ad Hoc Networks", Computer Science Department Carnegie Mellon University Pittsburgh, PA 15213-3891, http://www.monarch.cs.cmu.edu/

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

[5] G. F. Marias[1], Y, P. Georgiadis[1], D. Flitzanis[2] and K. Mandalas[2]," Cooperation enforcement schemes for MANETs: A survey", WIRELESS COMMUNICATIONS AND MOBILE COMPUTING,Wireless Communication. Mob. Computing. 2006; 6:319–332, Published online in Wiley Inter Science.

[6] Chinmay K. Nayak [1], G K Abani kumar[2], Parida[3], Das[4], "Detection of Routing misbehavior in MANET with 2ACK scheme ,Vol. 2, No. 1, Jan 2011

[7] R.Balakrishna[1], M. Muralimohan Reddy[2], U. Rajeswar Rao[3], G.A.Ramachandra[4], "Detection of Routing Misbehavior in MANET", using 2ACK ",IEEE International Advance Computing Conference (IACC 2009), Patiala, India, 6–7 March 2009

[8] Sheng Zhong , Jiang Chen , Yang Richard Yang , "Sprite: A Simple, Cheat-Proof, Credit-Based System for Mobile Ad-Hoc Networks" 2003 IEEE

[9] Marco Conti, Enrico Gregori, and Gaia Maselli, "Towards Reliable Forwarding for Ad Hoc Networks", IST-2001-38113 MOBILEMAN project

[10] Kashyap Balakrishnan, Jing Deng, Pramod K. Varshney,"TWOACK: Preventing Selfishness in Mobile AdHoc Networks", IEEE 2005

[11], YIH-CHUN HU∗ and ADRIAN PERRIG, "Ariadne: A Secure On-Demand Routing Protocol for Ad Hoc", Wireless Networks 11, 21–38, 2005

[12] "Java: The Complete Reference" (Herbert Schildt), Tata McGraw-Hill, 7th Edition.