ISSN:0975-9646

A Reputation Based Scheme to Prevent Routing Misbehavior in Manets

M. Sreedevi,
SVU College of CM & CS, SVUniversity, Tirupati

Abstract-Mobile Ad Hoc Network (MANET) system is not only subject to most of the well known attacks and threats that the wired and conventional wireless network suffered, but also a large number of additional attacks and threats. Security has become a primary concern in order to provide protected communication between mobile nodes in a hostile environment. Security is not a single layer issue but a multilayered issue. An individual mobile node may attempt to benefit from other nodes, but refuse to share its own resources. Such nodes are called selfish or misbehaving nodes and their behavior is termed selfishness or misbehavior. Most of the existing solutions to such misbehavior rely on the watchdog technique, which suffers from many drawbacks like that they may fail to detect misbehavior, causes additional routing overhead or raise false alarms in the presence of ambiguous collisions, receiver collisions, and limited transmission power. To mitigate the problem of routing layer misbehavior we analyze the existing misbehavior detection schemes and this paper propose a novel approach to overcome the drawbacks in existing techniques, and to improve the efficiency in detection.

Keywords: Mobile Ad Hoc Network, selfishness, Network security

INTRODUCTION:

There are two main types in MANETS: Closed and open. In an *open* MANET anyone is free to enter or leave the network (e.g., in airports and university campuses), whereas in a *closed* MANET only designated nodes may gain access (e.g., in a military setting). The network may be *Hierarchical*, where nodes have different roles and privileges and provide different services (e.g., network administration, certifying authority (CA), security and access control), or it may be fl*at*, where each node provides the same type of services (e.g., packet forwarding). Furthermore, the nodes may belong to a Single Administrative Domain (SAD), where only one administrator controls the network (e.g., the US army), or Multiple Administrative Domains (MAD), where there are multiple administrators (e.g., the US and UK armies in a joint operation)

Mobile Ad Hoc Network (MANET) system is not only subject to most of the well known attacks and threats that the wired and conventional wireless network suffered, but also a large number of additional attacks and threats. Security has become a primary concern in order to provide protected communication between mobile nodes in a hostile environment. Security is not a single layer issue but a multilayered issue. It requires a multi fence security solution that provides complete security spanning over the entire protocol stack. The study of this important issue reveals that security is divided into different directions of the work like secure routing, key exchange, distribution and management, secure architecture, intrusion detection and protection etc.

Mobile Ad Hoc Network is an emerging research area with practical applications. A Mobile Ad Hoc Network is a temporary infrastructureless network, formed by a set of self organized mobile hosts that dynamically establish their own network without relying on any central administration. In Mobile Ad Hoc Network, each mobile node acts as a router and forward packets for the nodes to achieve the multi-hop communication between the nodes. Thus communication in mobile ad-hoc networks functions properly only if the participating nodes cooperate in routing and forwarding. Security in Mobile Ad Hoc Networks is difficult to achieve, because of its basic characteristics such as open medium, dynamic topology, constrained resources and limited physical protection of nodes. However, performing network functions consumes energy and other resources. The limitation in energy resources along with the multi-hop nature of Mobile Ad Hoc Network causes a new vulnerability, i.e. packet dropping, which is caused either by malicious or selfish nodes. To save its energy a node may behave selfishly. However, whether for selfish or malicious reasons, a node may fail to cooperate during the network operations or even attempt to disturb the whole network functionality and can severely degrade the network performance, both of which have been recognized as misbehaviors. These nodes must be identified and excluded from the cooperative part of the network, as they only consume resources and don't contribute to the infrastructure.

Recently, this challenging problem received more and more attention among researchers, and some solutions have been proposed. Most of the existing solutions to such misbehavior rely on the watchdog technique, which suffers from many drawbacks like that they may fail to detect misbehavior, causes additional routing overhead or raise false alarms in the presence of ambiguous collisions, receiver collisions, and limited transmission power. To mitigate the problem of routing layer misbehavior we analyze the existing misbehavior detection schemes and need to propose a novel approach to overcome the drawbacks in existing techniques, and to improve the efficiency in detection.

RELATED WORK:

The problem of detecting misbehaving links instead of misbehaving nodes.

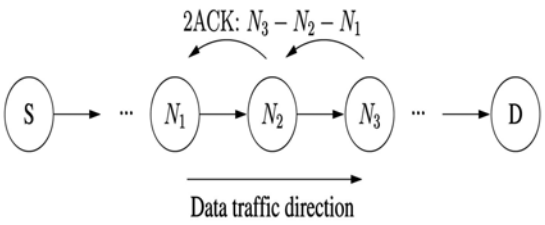

Fig. 1. The 2ACK scheme.

In the next-hop link, a misbehaving sender or a misbehaving receiver has a similar adverse effect on the data packet. It will not be forwarded further. The result is that this link will be tagged. Our approach discussed here significantly simplifies the detection mechanism.

Details of the 2ACK Scheme

The 2ACK scheme is a network-layer technique to detect misbehaving links and to mitigate their effects. It can be implemented as an add-on to existing routing protocols for MANETs, such as DSR. The 2ACK scheme detects misbehavior through the use of a new type of acknowledgment packet, termed 2ACK. A 2ACK packet is assigned a fixed route of two hops (three nodes) in the opposite direction of the data traffic route. Fig. 1 illustrates the operation of the 2ACK scheme. Suppose that N1, N2, and N3are 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 N3receives the data packet successfully or not. Such an ambiguity exists even when there is no misbehaving nodes.

The problem becomes much more severe in open MANETs with potential misbehaving nodes. The 2ACK scheme requires an explicit acknowledgment to be sent by N3to notify N1 of its successful reception of a data packet: When node N3receives the data packet successfully, it sends out a 2ACK packet over two hops to N1 (i.e., the opposite direction of the routing path as shown), with the ID of the corresponding data packet. The triplet

[N1--> N2--> N3] is derived from the route of the original data traffic. Such a triplet is used by N1 to monitor the link N2--> N3. For convenience of presentation, we term N1 in the triplet [N1--> N2--> N3] the 2ACK packet receiver or the observing node and N3the 2ACK packet sender.

Such a 2ACK transmission takes place for every set of triplets along the route. Therefore, only the first router from the source will not serve as a 2ACK packet sender. The last router just before the destination and the destination will not serve as 2ACK receivers.4 To detect misbehavior, the 2ACK packet sender maintains a list of IDs of data packets that have been sent out

but have not been acknowledged. For example, after N1 sends a data packet on a particular path, say [N1--> N2--> N3]in Fig. 1, it adds the data ID to LIST (refer to Fig. 2, which illustrates the data structure maintained by the observing node), i.e., on

N_2	N_3	C_{pkts}	C_{mis}	LIST
Next Hop Receiver	Second Hop Receiver	Packets Transmitted	2ACK packets Missed	List of data packet IDs

Fig.2. Data structure maintained by the observing node.

its list corresponding to N2--> N3.A counter of forwarded data packets, Cpkts, is incremented simultaneously. At N1, each ID will stay on the list for _ seconds, the timeout for 2ACK reception. If a 2ACK packet corresponding to this ID arrives before the timer expires, the ID will be removed from the list. Otherwise, the ID will be removed at the end of its timeout interval and a counter called C_{mis} will be incremented. When N3receives a data packet, it determines whether it needs to send a 2ACK packet to N1. In order to reduce the additional routing overhead caused by the 2ACK scheme, only a fraction of the data packets will be acknowledged via

2ACK packets. Such a fraction is termed the acknowledgment ratio, Rack. By varying Rack, we can dynamically tune the overhead of 2ACK packet transmissions. Node N1 observes the behavior of link N2--> N3for a

period of time termed Tobs. At the end of the observation period, N1 calculates the ratio of missing 2ACK packets as Cmis=Cpkts and compares it with a threshold Rmis. If the ratio is greater than Rmis, link N2-->N3 is declared misbehaving and N1 sends out an RERR (or the misbehavior report) 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--> N3as 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. The 2ACK scheme can be summarized in the pseudocode provided in the appendix for the 2ACK packet sender side (N3) and the observing node side (N1).

PROPOSED WORK:

The watchdog detection has a very low overhead. Unfortunately, the watchdog technique suffers from several problems such as ambiguous collisions, receiver collisions, and limited transmission power. The main issue is that the event of successful packet reception can only be accurately determined at the receiver of the next hop link, but the watchdog technique only monitors the transmission from the sender of the next-hop link. Noting that a misbehaving node can either be the sender or the receiver of the next-hop link, we focus on the problem of detecting misbehaving links instead of misbehaving nodes. In the next-hop link, a misbehaving sender or a misbehaving receiver has a similar adverse effect on the data packet. It will not be forwarded

further. The result is that this link will be tagged. Our approach discussed here significantly simplifies the detection mechanism.

Coming to 2ACK getting acknowledgments in inverse order. If any middle node will be misbehavior means it arises two issues that is first one is sending acknowledge which received by previous node it can't forward to before node. Second issues are even if the packets received by the correct node only by the misbehavior of middle nodes source node realizes packets are not reached to the required nodes. Memory overheads are other issues in 2ACK. how it happens means every node should have capability to store all acknowledgements which coming from precede nodes if node having less or any technical problem in memory obviously there is chance of losing acknowledgements.

In this proposed technique instead of getting the acknowledgements for every random number in reverse order, detecting which nodes are actual misbehavior. It is basically introduced as an add-on to existing routing protocols for MANETs, such as DSR (Dynamic Source Routing). After that implementing the count values in nodes for IN FLOW of packets and OUT FLOW of packets. Inspections the equality of count number in nodes, if any shows the inequality number then dynamically deposits a threshold value. Here the threshold is of two

Values one is inflow threshold value and second is outflow threshold value. When ever the threshold value is exceeded. Then the nodes are in misbehavior.

HOW IT WORKS:

Let us consider example and illustrates for better understanding. Fig. 3 shows that the route discovery. After routing discovery the source node starts that the sending of packets to the destination node with the help of the intermediate nodes which are dynamically prepared

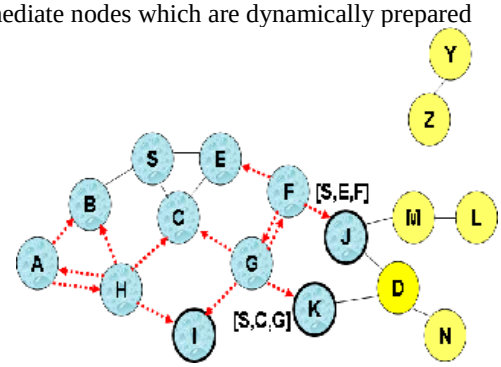

Fig.3 Route Discovery

topology, whenever The source node starts the sending of packets to the intermediate nodes. Then at the intermediates nodes dynamically a new data structure is created with two variables namely inflow and outflow. Inflow measures the how many packets incoming to the node by the pervious node and outflow measures the how many packets are dropping to the proceed node. These two count values (inflow and outflow) are equal then the nodes are working in discipline manner. Other wise they are not. In two steps these misbehavior problem is solved. First step is, When ever the inflow and outflow value get unequal then dynamically a message is send to the source node. This message contains the values of inflow and outflow count values. Source node identified that which value is in low. Whether inflow count is low or outflow count is low. Then the source node search for another infrastructure internal.

Second step is two thresholds values are born in which the node getting inequality. These threshold values is used to compare with the inflow count and outflow count when ever the inflow and outflow values are decreases compare to threshold value obviously it send a message to the source node. This node is misbehavior. Then the source node will change the topology to accomplishment remaining task.

For better illustrates of this proposed system. By considering the Fig.2. Node S the source node which is already discovery its path by DSR. According to the route which it is discovery it sends the packets S→F→J. assume that the J is misbehavior node. Based on proposed technique the inflow and outflow values are inequality. It sends a message to the source node. Internal source node search for another topology when ever the message is received. After a few seconds node J is exceed the threshold values then it send another message to the source node S. then it go through the new infrastructure.

ADVANTAGES:

- No memory overhead form due the messages
- There are no ambiguous collisions and receiver collisions.
- No exceed of acknowledgments during transmission of packets.
- With in single step source node the status of two nodes namely next node and previous node.
- It deals with the reliable transfer of file from source to destination. The file needs to be stored at source for certain amount of time even if it has been transmitted. This will help to resend the file if it gets lost during transmission from source to destination.
- Proposed scheme is its flexibility to Control overhead with the use of the Rack parameter.

CONCLUSION

The proposed system is that detects misbehaving links in Mobile Ad Hoc Networks. By knowing the Inflow and outflow of the packets between the nodes. This technique detects and prevents the misbehavior nodes. When ever the scheme knows that in the dynamically topology having maximum nodes are misbehavior. It changes to the new topology.

REFERENCES:

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

[2] H.Miranda and L.Rodrigues, "Preventing Selfishness in Open Mobile Ad Hoc Networks", October 2002.

[3]. Liu J, Issarny V. "Enhanced reputation mechanism for mobile ad hoc networks". In Proceedings of 2nd International Conference on Trust Management, March 2004.

[4]. K. Balakrishnan, J. Deng, and P.K. Varshney, "TWOACK: Preventing Selfishness in Mobile Ad Hoc Networks," Proc. IEEE Wireless Comm. and Networking Conf. (WCNC '05), Mar. 2005.

[5].Kejun Liu, Jing Deng, Pramod K. Varshney, Kashyap Balakrishnan "An Acknowledgement-Based Approach for the Detection of Routing Misbehavior in MANETs", IEEE Transactions on Mobile Computing

vol.6,No.5,May 2007. [6]. Song JianHua, Ma ChuanXiang "A Reputation-based Scheme against Malicious Packet Dropping for Mobile Ad Hoc Networks" , 2009 IEEE.

[7]. Alper T. M. zrak, Keith Marzullo, "Detecting Malicious Packet Losses", IEEE Transactions on Parallel and Distributed Systems, vol. 20, no. 2, February 2009.

AUTHOR :

M. Sreedevi M.CA,M.Phil,(Ph.D)
Email-id :msreedevi_svu2007@yahoo.com
Currently working as a Assistant Professor, SVU College of CM&CS, Department of Computer Science, S.V.University, Tirupati, A.P.