Security Enhancement in MANETS using the Concept of Leader Nodes

Jatinder Singh
Associate
Sapient Global Markets, Gurgaon,India

ABSTRACT

This paper is aimed to detect the malicious behavior of the nodes by first selecting some set of leader nodes. The leader nodes are those set of nodes in a network which monitor the whole network of nodes. The set of leader nodes is referred as the dominating set. A set is dominating if all the nodes of the network are either in the set or neighbors of nodes in the set. The efficiency of a commanding-set-based broadcasting or routing is mainly dependent on the overhead in constructing the dominating set and the size of the dominating set. The leader nodes are working in the promiscuous mode and they overhear traffic in its neighborhood. In promiscuous mode the system (NS-2) by-pass the MAC(Media Access Control) filtering procedure and the nodes working in promiscuous mode, receives the packets that are sent or received by any other node in its transmission range even if the packets were not intended for that node, this phenomenon is commonly known as packet overhearing. The leader nodes also monitor the topology of the network and reports if there are any changes in the network on the basis of a certain criteria, moreover they also detect if any packet is modified, and the snooping attack and calculate the forward packet ratio.

Keywords:

DSR;Malicious;Leader;security;Mobile Ad Hoc Networks; routing protocols

1. INTRODUCTION

1.1 Mobile Ad Hoc Networks

Ad hoc networks are combination of mobile links without existence of any centralized control or pre-existing infrastructure. Such kind of networks generally use multihop paths and wireless radio communication channel. Wireless networks provide facility to transmit/receive data among users in a common area. Thus, communication between nodes is established by multihop routing.

Movements of nodes in a mobile ad hoc network cause the nodes to move in and out of range from one another. As the result, there is a continuous making and breaking of links in the network, making the network connectivity (topology) to vary dynamically with time.

Because of this time-varying nature of the topology of mobile ad hoc networks, traditional routing techniques, such as the shortest-path and link-state protocols that are used in fixed networks, cannot be directly applied to ad hoc networks. A

Fundamental quality of routing protocols for ad hoc networks is that they must dynamically adapt to variations of the network topology.

1.2 DSR Protocol

DSR is a simple and efficient routing protocol designed specifically for use in multi-hop wireless ad hoc networks of mobile nodes. DSR allows the network to be completely self-organizing and self-configuring, without the need for any existing network infrastructure or administration. Dynamic Source Routing is used as the principal protocol for selection of malicious nodes. In the proposed method, only few nodes are required to function in promiscuous mode, this results in the cutback of network overhead as compared to the protocols which require all the network nodes to work in promiscuous mode. Nodes working in the promiscuous mode overhear all the transmissions within its range; this requires each node to have high energy capacity. So selecting a set of nodes called Monitor Nodes (MN) set using [4] such that all the nodes in the network are either in the MN set or the neighbors of the nodes in the MN set. The nodes which belong to the MN set are called monitor nodes. Monitor nodes always operate in promiscuous mode and detect the malicious nodes in its neighborhood. The rest of the nodes in the network which are not in the MN set are called regular nodes. The monitor nodes themselves are also monitored by the neighboring monitor node/s and they can also be caught in case they misbehave.

The DSR protocol is composed of two mechanisms that work together to allow the discovery and maintenance of source routes in the ad hoc network:

1.2.1 Route Discovery is the mechanism by which a node S wishing to send a packet to a destination node D obtains a source route to D. Route Discovery is used only when S attempts to send a packet to D and does not already know a route to D.

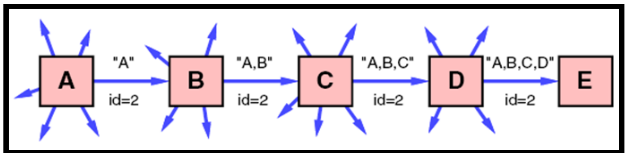


Fig.1 Route Discovery

1.2.2 Route Maintenance is the mechanism by which node S is able to detect, while using a source route to D, if the network topology has changed such that it can no longer use its route to D because a link along the route no longer works. When Route Maintenance indicates a source route is broken, S can attempt to use any other route it happens to know to D, or can invoke Route Discovery again to find a new route. Route Maintenance is used only when S is actually sending packets to D.

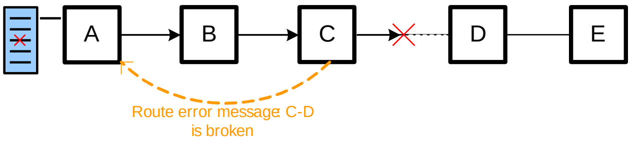


Fig. 2 Route Maintenance

This paper focuses on detection of two types of malicious behavior exhibited by the nodes, the first one is the malicious topology change behavior and the other one is the malicious packet drop behavior in DSR protocol. These attacks (or malicious behavior) hinder the communication between nodes and makes the routing process difficult. Hence, these need to be corrected as they are not handled by the standard Dynamic Source Routing protocol (DSR).

2. BACKGROUND AND LITERATURE SURVEY

Marti et al. designed Watchdog and Pathrater mechanism [1] to optimize the packet forwarding method in the Dynamic Source Routing (DSR) protocol [2]. It consists of two components: Watchdog and Path rater. The Watchdog detects selfish nodes that do not forward packets and the Path rater helps routing protocols to avoid these nodes. It assigns ratings to the nodes, based upon the feedback it receives from the Watchdog. These ratings are then used to select routes having nodes with the highest forwarding rate. Watchdog's weaknesses are that it might not detect a misbehaving node in the presence of: Ambiguous collisions, Receiver collisions, Limited transmission power, false misbehavior and Partial dropping.

In technique [3] the algorithm performs routing checks on incoming packets. Filters are placed at key points in a network, unlike perfect ingress filtering which places filters at every node. Implementing this would require modifying some or all of the nodes in the network. Watchdog [1] is a technique where each node snoops the retransmission of every packet it forwards. If the watchdog detects that the node has not correctly retransmitted the packet, it can raise a warning. Again this requires modification of some or all of the nodes in the network and is developed primarily for ad hoc networks operating with Omni-directional transmissions. As known from the facts that encryption and message integrity checking comes at a price of increased computation overheard at multiple nodes. The method described in [4] introduces an approach that can be applied to calculate the trust in a dynamic way, and also protect the message against modification, without significantly increasing the overheads.

Most of the work on routing security focus on the efficient use of digital signatures or shared secret keys to authenticate and confide the data and routing headers. However, they always tend to find the shortest path between source and destination irrespective of the presence of malicious nodes in between. To overcome this problem and in quest for a trusted routing solution, the technique proposed in [5] is capable of finding secure end-to-end paths and can also prevent any attack from colluding malicious nodes.

In a DSR [2] based network, each transmitted data packet contains the complete list of node addresses that the packet has to traverse in order to reach its final destination. Intermediate nodes blindly forward these packets as per the attached list without taking into consideration the behavioral pattern of the subsequent nodes. In the paper presented in [6], describes variant of the DSR protocol in which intermediary nodes act as Trust Gateways. These gateways take into account the contemporary trust levels of the network nodes and thus facilitate in detecting and evading malicious nodes.

In fixed networks, trust infrastructures like Certification Authorities and Key Distribution Centers are generally used to provide default trust relationships. However, the creation of such an entity in an ad hoc network is considered neither feasible nor pragmatic. In “TRUST-BASED ROUTING FOR AD-HOC WIRELESS NETWORKS**”** [7], a novel mechanism for establishing trust based routing in ad-hoc networks without necessitating a trust infrastructure. The author accentuate, that the proposed mechanism is most suitable for ad hoc networks that can be created on the fly without making any suppositions or imposing pre-configuration requirements.

The Dynamic Source Routing (DSR) protocol [2] is one such protocol that helps to create and maintain routes in an ad-hoc network in spite of the dynamic topology. The accurate execution of this protocol requires sustained benevolent behavior by all participating nodes in the network. This is generally not possible to achieve and so a number of attacks may be launched against the DSR protocol, which lead to the malfunction of the network usually at a critical point in time. A novel technique of discovering and maintaining dependable routes in an ad hoc network even in the presence of malicious nodes has been described in [8].

In DSR [2], nodes are subjected to a variety of attacks by other nodes. These attacks range from naive passive eavesdropping to vicious battery draining attacks. Routing protocols, data, battery power and bandwidth are the common targets of these attacks. In order to overcome such attacks a number of routing protocols have been devised that use cryptographic algorithms to secure the routing mechanism, which in turn protects the other likely targets. A limiting requirement regarding these protocols is the reliance on an omnipresent, and often omniscient, trust authority. This reliance on a central entity is against the very nature of ad-hoc networks, which are supposed to be improvised and spontaneous. The method presented in the paper [9] is, a trust-based model for communication in ad hoc networks that is based on individual experience rather than on a third party advocating trust levels. The model introduces the notion of belief and provides a dynamic measure of reliability and trustworthiness in pure ad-hoc networks.

Security is a means of creating trust in a link or route using a variety of methods. Usually these methods make use of cryptographic mechanisms to enforce security. In such schemes, the trust is usually placed in the strength of the scheme and the length of the encryption key. Although secure, these schemes only provide two views of trust, by either the presence of security or its absence. In the paper [10], a unique way of locating and preserving dependable routes in ad hoc networks that execute the DSR protocol [2]. Instead of using hard security mechanisms it can employ an effort-return trust model that is influenced by the human behavior model. The aim of the trust model is to establish dependable routing in ad hoc networks without necessitating a centralized or distributed trust infrastructure. The trust levels are associated to network nodes so as to compute trustworthy routes through the network. With the help of extensive simulations, it is shown that the scheme enhances the throughput and lowers the packet loss of the network in the presence of malicious nodes. These malicious nodes may carry out a number of modification attacks against the network including the creation of grey and black holes.

The main threat for routing in a MANET is the existence of selfish and malicious nodes. The goal of a selfish node is to maximize its own welfare; on the other hand a malicious node tries to prevent the network from operating efficiently or properly. Without any countermeasures against these threats, the network performance decreases considerably. The paper [11] propose a secure and efficient routing scheme using a game theoretical approach and trust relationships between the nodes that assumes a ‘‘Dynamic Bayesian Game” model [12] among the nodes to find the optimal strategies of legitimate and malicious nodes. Moreover, using the ‘‘watchdog” technique [1] and the ‘‘acknowledgement” mechanism (ACK), it can construct trust relationship between the nodes.

3. EXPLANATION OF FLOWCHART FOR THE LEADER NODE SELECTION

The algorithm first compute the node degree of all the wireless nodes then, it takes a particular node and checks the circular links for that node, by the checking of circular links it mean that the node arranges its neighbors in increasing order of their node ID’s and checks if that particular set of nodes forms circular links or not (either connected by 1 hop or 2 hops), if all the circular links are not present then the node is straight away marked as the leader node, otherwise it then checks for the log links , and for that it take the floor values of the node degree for that node, supposedly it comes “n” then from it check the connectivity of nodes (either connected by 1 hop or 2 hops) at a distance of n hops away from that node in the circular fashion that was previously taken. If all the log links are present then the node is marked as the non-leader node and if even one link is missing then it is marked as the leader node.

3.1 Flowchart for Malicious Node Detection

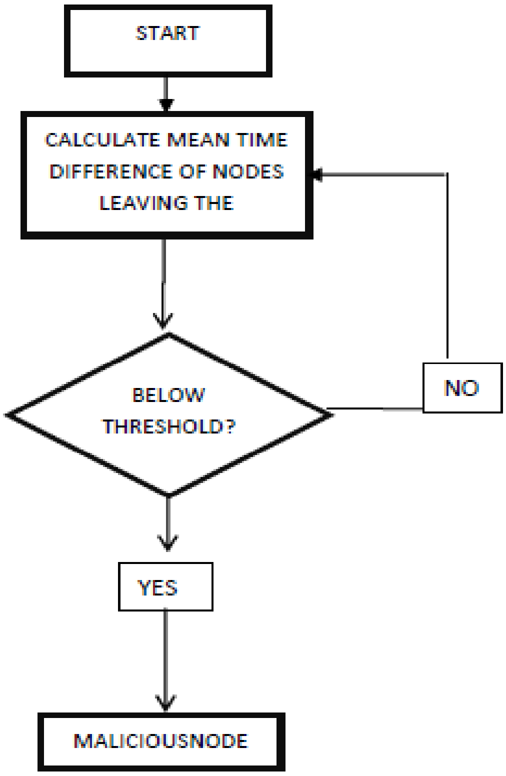


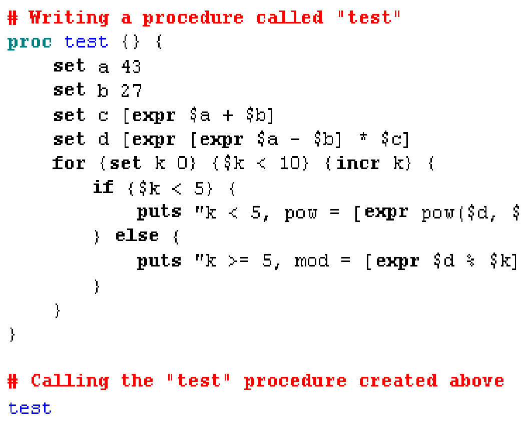
Fig.3 Flowchart for Malicious Detection of node

3.1.1 *Explanation of Flowchart for the Malicious Node Detection*

The task of malicious node detection is accomplished by running two algorithms in parallel (See Fig. 3). It calculates the mean on the time difference of the nodes leaving the network, and this is achieved by storing the timestamp values whenever the node goes out of the field of the leader node.

4. IMPLEMENTATION

Example of an OTCL procedure



```
# Writing a procedure called "test"
proc test {} {
    set a 43
    set b 27
    set c [expr $a + $b]
    set d [expr [expr $a - $b] * $c]
    for {set k 0} {$k < 10} {incr k} {
        if {$k < 5} {
            puts "k < 5, pow = [expr pow($d, $k)]"
        } else {
            puts "k >= 5, mod = [expr $d % $k]"
        }
    }
}

# Calling the "test" procedure created above
test
```

Fig 4 OTCL Procedure

- Proc: define a procedure, followed by an procedure name and arguments
- Set: assign a value to a variable
- [expr …]: to make the interpreter calculate the value of expression within the bracket
- To assign to variable x the value that variable has, then write “set x $a”.
- Put: prints out

4.1 Schematic View of the Project

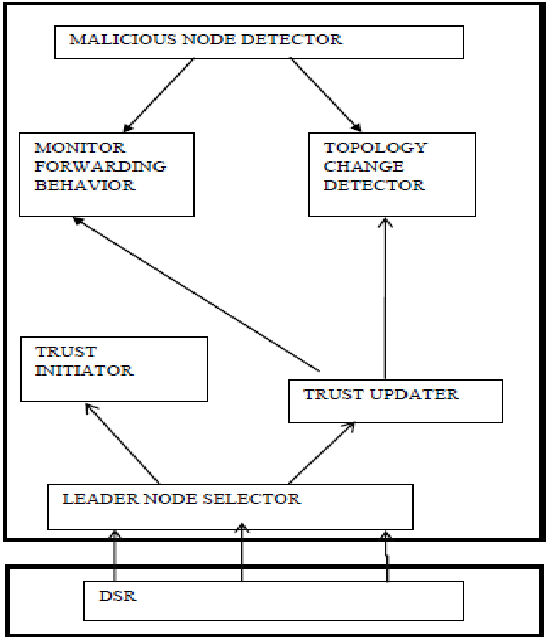


Fig.5 Schematic View

The previous diagram depicts the schematic view of the paper. The model for malicious detection is built on the standard DSR protocol. The leader node selector module selects a set of leader nodes from the network depending upon the network topology and the number of nodes in the network. Each node in the network is monitored by one or more leaders.

The leader node provides initial trust values to all the nodes in the network and the trust updater module updates the trust values of nodes in the network based on the forwarding behavior and topology change factor, based on the mobility of the mobile nodes.

The malicious node detector module detects the malicious nodes based on certain threshold value of trust and ensures that they do not take part in the routing process. It is not reasonable to construct a general method for updating trust values that will be applicable to all applications in all domains. The function designed aims to function in domains with several malicious nodes.

A function for updating trust can depend on several parameters. In the list below some of the possible parameters are listed.

- Previous trust values.
- Lowest/Highest trust value ever assigned.
- Number of positive/negative experiences in the past.
- The situation/value of an experience.

Ideally the behavior of the function should depend on the expected trust dynamics in a given situation.

4.2 Neighbor Monitoring

4.2.1 Neighbor Remove Table

Table 1 Neighbor Removal Table

Node Address	Time of Leaving the Network(in seconds)	Time Difference
X	T1	t0=0
X	T2	t1=T2-T1
X	T3	t2=T3-T2
X	T4	t3=T4-T3

The mean value m is given by m= (t0 + t1 + t2 + t3) / 4
If m is found lower than a threshold value then the node is identified as a malicious node.

4.2.2 Experimental Values in Neighbor Remove Table

Table 2 Experimental Values

Node Address	Time of Leaving the Network(in seconds)	Time Difference
2	10.77	0
2	20.77	10.77
2	30.77	10.77
2	40.77	10.77

Mean m = 30 / 4 = 7.5

4.3 Algorithm for Selection of Leader Nodes

4.3.1 Algorithm

```
begin
my status = nonforward;
r = my degree;
ifr> 1
i = 0;
s = 1;
while(s ≤ r)
while(i< r) and (my status = NON LEADER)
j = (i + s) mod r;
x = my neighbor id[i];
y = my neighbor id[j];
if((x, y) _∈E) and (_ ∃z s.t. z.id > my id
and (x, z) ∈E and (z, y) ∈E)
/* require 2-hop information */
my status = LEADER;
exit;
endif
i++;
endwhile
s = 2s;
endwhile
endif
end.
```

4.3.2 Explanation

For each node v that has more than one neighbor, the algorithm first arranges its neighboring nodes in a total order, for example, an increasing order of *node id*s. Let the neighboring nodes of v listed in this order be v0, v1…, vr−1, where r = deg(v). The algorithm checks the pairs of nodes (vi, v(i+s)mod r), where i = 0, 1, . . . r − 1 and s = 2j, j = 0, 1, . . . , _log2 r_. If there exists a pair of nodes that are neither connected directly nor connected via a node u that has a higher priority than v then v is marked as leader node.

The distributed algorithm runs in O(d log d) time for 1-hop connectedness and O(d2 log d) for 2-hop connectedness, respectively. Previous algorithms for 1-hop and 2-hop connectedness run in O(d2) and O(d3), respectively.

The proposed distributed algorithm for each node v is shown in Algorithm 1. It has used my id and my degree to denote node v and deg(v), respectively. In the algorithm, my neighbor id, an array of length deg(v), stores the *id*s of v's neighbors. The output of the algorithm is my status that will be "Leader" or "Non Leader".

5. RESULTS

5.1 Simulation Parameters

Table 3 Simulation Parameters

Simulation parameters	Values
Nodes	8-250
Routing protocol	DSR
MAC Layer	IEEE 802.11
Traffic model	CBR(Constant bit rate)
Packet Type	TCP(Transmission Control Protocol)
Area	500 X 600
Transmission range	250 m
Transmission threshold power	0.28183815
Txpower	0.173
Rxpower	0.05
Packet size	512 bytes
CS range	550 m

5.2 Graphs

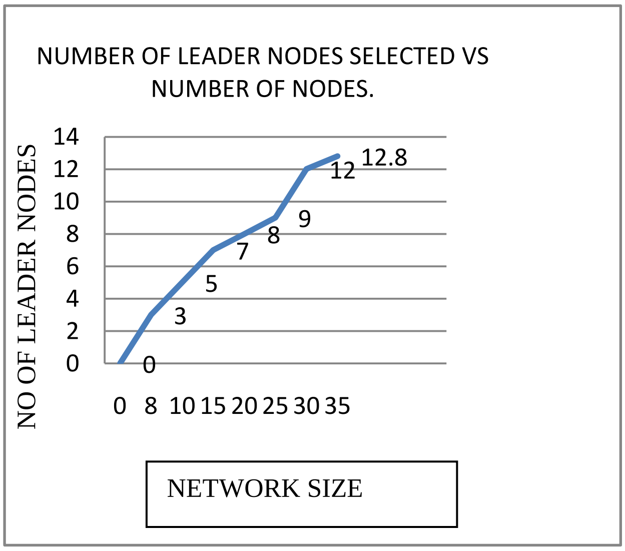


Fig. 6 Results

5.2.1 Explanation

The above graph is a plot of number of leader nodes against the network size and as per the simulations the relationship between them is monotonically non-linear. One thing which is certain is that as the network size increases the number of leader nodes as selected by the algorithm discussed in section 4.4, increases.

6. CONCLUSION

In this paper work the author have proposed a novel method for the detection of malicious nodes in DSR protocol this method is named as MD-DSR (Malicious Detection In Dynamic Source Routing) which detects malicious nodes. The detection of malicious nodes is done by leader nodes these leader nodes monitor all the nodes in its neighborhood.

Through simulations done in NS2 the results show that the proposed method MD-DSR is better than the standard DSR in Terms of packet delivery ratio, throughput, etc. The modified version of it detects malicious nodes which was previously not there, moreover it detects the malicious nodes based on the two different and exclusive criteria first is the mobility factor and the other one is the forward packet ratio.

7. REFERENCES

[1] S. Marti, T. J. Giuli, K. Lai, and M. Baker, ”Mitigating routing misbehavior in mobile ad hoc networks,” in the Proceedings of Sixth Ann.Int’l Conf. Mobile Computing and Networking (MobiCom), pp.255-265, 2000.

[2] D. B. Johnson and D. A. Maltz, ”Dynamic Source Routing in Ad-Hoc Wireless Networks,” Mobile Computing, T. Imielinski and H. Korth, Eds., Kluwer, pp. 153-181, 1996.

[3] K.Park and H.Lee, On the effectiveness of route based packet filtering for distributed D DoS prevention in power law internets, In proceedings of ACM SIGCOMM, pages 15-26, 2001.

[4] S. K. Dhurandher and V. Mehra.Multi-path and message trust-based secure routing in ad hoc networks. Proc. Int. Conf. Advances in Computing, Control and Telecommunication Technologies (ACT 2009), Trivandrum, India (Dec. 28-29, 2009), 189-194.

[5] T. Ghosh, N. Pissinou and K. Makki.Towards Designing a Trusted Routing Solution in Mobile Ad Hoc Networks. Mobile Networks and Applications, Springer Science, 10, 2005, 985-995.

[6] A.A. Pirzada and C. McDonald.Deploying trust gateways to reinforce dynamic source routing. Proceedings of the 3rd International IEEE Conference on Industrial Informatics, IEEE Press, 2005, 779-784.

[7] A.A. Pirzada, A. Datta, C. McDonald. TRUST-BASED ROUTING FOR AD-HOC WIRELESS NETWORKS. IEEE, 2004, 326-330.

[8] A.A. Pirzada and C. McDonald.Dependable Dynamic Source Routing without a Trusted Third Party. Journal of Research and Practice in Information Technology, Vol. 39, Issue 1 (February 2007).

[9] A.A. Pirzada and C. McDonald.Trust Establishment In Pure Ad-hoc Networks. Wireless Personal Communications, Springer, 37, 2006, 139-163.

[10] Asad Amir Pirzada, AmitavaDatta, Chris McDonald. Incorporating trust and reputation in the DSR protocol for dependable routing. Computer Communications, Vol. 29, Issue 15 (5 September 2006), 2806-282.

[11]E. Ayday, F. Fekri. A protocol for data availability in Mobile Ad-Hoc Networks in the presence of insider attacks. Ad Hoc Networks, Vol. 8, Issue 2 (March 2010), 181-192.

[12] D. Fudenberg, J. Tirole, Game Theory, The MIT Press, Cambridge,MA,1991.

[13] Yamin Li, Shietung Peng, Wanming Chu. An Efficient Algorithm forFinding an Almost Connected Dominating Set of Small Size on WirelessAd Hoc Networks.IEEE, 2006.