Volume 2, No. 2, Mar-Apr 2011

ISSN No. 0976-5697

International Journal of Advanced Research in Computer Science

RESEARCH PAPER

Available Online at www.ijarcs.info

A Novel Approach for Identifying Selfish Nodes by Trust Levels Using Fuzzy in Manet

Vijayan R*
Assistant Professor (senior)
School of Information Technology and Engineering,
VIT University,
Vellor, Tamil Nadu, India
vijayan_ram@yahoo.com

Mareeswari V
Assistant Professor
School of Information Technology and Engineering,
VIT University,
Vellore , Tamil Nadu, India
marees_venkat@yahoo.co.in

Ramakrishna K
M.Tech in Information technology [Networking]
School of Information Technology and Engineering,
VIT University,
Vellor, Tamil Nadu, India
ramakrishna87@gmail.com

Abstract: Mobile Ad Hoc Networks (MANETs) is a wireless network with mobile nodes. These networks are become more vulnerable because their features of open medium, topologies no centralized monitoring agents. In MANET cooperation is necessary between the nodes to transfer the packets out of their propagation. Cooperating nodes must trust each other. There are several ways to calculate the trust among those nodes. But some of them not considered the energy factor. In this paper, a novel approach is proposed to trust the nodes in MANET. The decision on nodes depending on this trust value is done by fuzzy logic. The entire this frame work is to detect the selfish nodes mobile ad hoc network.

Keywords: MANET, Misbehavior, Trust management, fuzzy logic, Energy utilization factor, selfish nodes.

I. INTRODUCTION

Mobile ad-hoc networks (MANETs) have been receiving a lot of attention during the past few years because the rapid expansion of mobile devices and the interest in mobile communication. A MANET is an autonomous system of mobile nodes connected by wireless links. Each node operates not only as an end-system, but also as a router to forward packets. The nodes are free to move about and organize themselves into a network. MANETs do not require any fixed infrastructure such as base stations. So, it is an attractive networking option for connecting mobile devices quickly and spontaneously, such as military applications, emergent operations, personal electronic device networking, and civilian applications like an ad-hoc meeting or an ad-hoc classroom.

MANETs have several salient characteristics, such as dynamic topologies, bandwidth constrained, variable capacity links, energy constrained operation, limited physical security. Because of these features, MANETs are particularly vulnerable to all kinds of attacks launched through compromised node. Unreliable wireless links are vulnerable to jamming and by their inherent broadcast nature facilitate eavesdropping. Constraints in bandwidth, computing power, and battery power in mobile devices can lead to application-specific trade-offs between security and resource consumption of the device.

All the nodes in MANET must co-operate with each other to route the packets. In general, uncooperative nodes in MANETs may be of two types: *malicious nodes* and *selfish nodes*. The nodes belonging to the first category are either faulty and therefore cannot follow a protocol, or are intentionally malicious and try to attack the system. A selfish node, on the other hand, is an economically rational node whose objective is to maximize its own welfare, defined as the benefit of its actions minus the cost of its actions. Because forwarding a message will incur a cost, a selfish node will need incentive for doing it.

II. RELATED WORK

The nature of the wireless and mobile environment makes it vulnerable to adversary's malicious attacks. Such networks are susceptible to attacks ranging from passive eavesdropping to active interfering. In MANETs nodes are receptive founding captured, compromised and hijacked because they are units capable of roaming independently. A two phase detection process has been proposed for the detection of the nodes that are not authorised to access the services and the nodes that have been compromised in MANET [1]. Found come as trustworthy, a node must make a reasonable effort to perform its generic functions and duties in the network in a dependable manner, broadly categorized under headings of Routing/Forwarding, Quality of Service, and Security [2]. A new approach has been proposed to bring out the complementary relationship between key distribution and misbehaviour detection for developing an integrated security solution for MANETs [3].

In [4] the authors proposed a trust model using fuzzy model, they explained fuzzy direct trust model, fuzzy indirect trust model, fuzzy recommendation trust model, and Fuzzy transitivity recommendation trust model. [5] In this paper, a backward fuzzy trusted routing algorithm based on fuzzy dynamic programming approach under mobile ad hoc network environment has been proposed. As a modification to the traditional Dynamic Source Route (DSR) protocol, we

have presented a Fuzzy Trust Dynamic Source Route protocol (FTDSR). In [6] a fuzzy based approach for calculating the trust levels of a node for detection of secure route to forward the packets. In this paper they considered the packet dropping, wrong routing, and replay attacks as the parameters for the calculation of trust level of a node. They also considered the reasons for packet dropping.

III. TRUST AND TRUST SYSTEMS

Trust, in general, is a directional relationship between two entities and plays a major role in building a relationship between nodes in a network [3]. In [2] trust represents the expectation that the participants will enforce the rules defined in the community specification (or doctrine) and that the membership of the community will be governed by clearly defined constraints. In [3] trust is defined as a firm belief in the competence of an agent to act dependably, securely, and reliably within a specified context. Trust leads to the notion of trusted systems. In [2] a trusted system is defined as an entity whose security mechanisms are isolated from and are uncircumventable by unauthorized users; the system can be identified, content controlled and secure, and managed by a competent authority. With respect to ad hoc networks, this essentially implies that each participating node has the necessary security components that offer the desired security services which cannot be overridden in an unauthorized way. Each node can then be trusted to perform networking related services (e.g., routing, forwarding) as well as end system services.

IV. DESIGN COMPONENTS

In this proposed scheme, every node in the network monitors the behaviour of its neighbours, and if any abnormal action is detected, it invokes an algorithm to determine whether the suspected node is indeed malicious. By 'neighbours' of a node, we mean all the nodes in the network that are one-hop distance from the node. The proposed mechanism builds trust in the network by interactions among some security components running each node as in Figure.

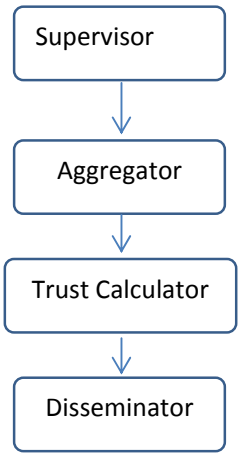

Figure: 1 Component

These components are:

- Supervisor
- Aggregator
- Trust calculator
- Disseminator

A. Supervisor

The supervisor module monitors neighbours by passively listen to their communication. For detecting packet drops, the supervisor module uses Passive Acknowledgement (PACK) mechanism that checks whether the neighbours really forward the packets or drops them. The collected data is audited by the monitor. The deviation from normal behaviour of a neighbour is used as an indicator for the unbiased degree of selfishness because this is independent of past behaviour of the neighbour node. If this unbiased deviation exceeds a pre-set threshold, the aggregator module is invoked.

B. Passive Acknowledgement

As Sonja Buchegger, C´edric Tissi`eres, and Jean-Yves Le Boudec mentioned in [7] PACK can be used for Route Maintenance when originating or forwarding a packet along any hop other than the last hop. PACK cannot be used with the last hop because it will never retransmit a packet destined to itself. PACK needs two conditions to be applied: nodes have their network interfaces in promiscuous mode, and network links operate bidirectional.

PACK works as follows. When a node receives a packet to be forwarded to a node other than last hop, the node sends the packet without requesting a network-layer acknowledgment (ACK). If it doesn't overhear the retransmission of the next node within a timeout, the node retransmits the packet again, without network-layer ACK request. After a certain number of trials, a network-layer ACK request must be used instead of PACK for all remaining attempts for that packet. When a node receives a new packet, it considers it as a
PACK if the following checks succeed:

[a] Source address, destination address, protocol identification and fragment offset fields in the IP header of the two packets must match.

[b] If either packet contains a DSR Source Route header, both packets must contain one, and the value in the Segments Left field (it indicates the number of hops remaining until the destination) in the DSR Source Route header of the new packet must be less than that in the first packet.

C. Aggregator

This module collects all the details of the communication that already happened. That can be used to calculate the number of packets dropped. Usually the output of this module is the trace file in NS2. All this collected data is considered as reputation of the nodes which is collected at the time of simulation. The accused node calculates the group's trust in its behaviour using the received values and broadcasts the computed group trust along with the received

responses to all the neighbours. All these messages are encrypted and time stamped to replay attacks. For computing group trust value from the received responses, fuzzy based scheme used.

D. Trust Calculator

In the proposed schema the trust level of a node is determined by the percentage of packet dropped. This percentage is treated as fuzzy input variable and the output of the algorithm is trust level of a node. When a node enters a network an initial trust level (based on the number of credentials of a node being the number of neighbour's identifying it as an authentic node) is assigned to a node. This trust level is checked and modified periodically to allow a node found a part of the network based on the current trust level of the node. If the trust level of a node is found low because of authentic reasons the node is tagged as a bad node and is not allowed to carry any further communications in the network. Based on these reasons finally the trust level of a node is evaluated.

E. Fuzzy Trust calculation

The proposed scheme is based on but different from existing work with enhanced components [8]. The trust management framework is made up of following three components [9]. First component is to calculate direct trust existing between the nodes with direct communication with that node that is called as Direct Trust agent (DTA), Second component is to the trust on a node with the help of another trusted node, that is called as indirect trust agent (IDTA) or recommendation agent (RA) and the third component aggregator (AG) uses the DTA and IDTA to calculate the total trust value of target node.

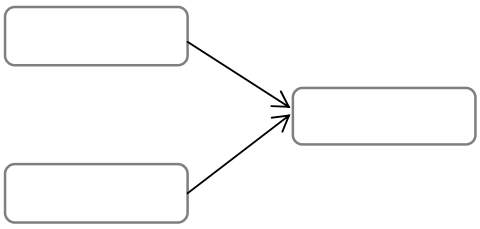

Figure2. Total trust calculation

F. Direct Trust Agent

Direct trust agent performs the following tasks derivation of trust, quantification and trust computation. Node x want to calculate the trust value on node y termed as dt_{xy}. This dt_{xy} is a function of S, F and E i.e.

$$t_{xy} = K(S, F, E) \quad (1)$$

here S is a function of successful communication metrics, F is a function of not successful communication metrics and E is energy utilisation factor as explained in section 4.3.1.4. To calculate the direct trust on node y, node x has to monitors the following statistics.

Data Packets forwarded s_1, Control packets forwarded s_2, Data packets received s_3, Control packets received s_4.All the above statistics are successful communication statics. Packets dropped f_1, packets dropped due to unknown reason f_2, packets forwarded delay f_3, packets misrouted f_4 and all these are statistics of failure communication. Now we can rewrite S and F as

$$S = P(s_1, s_2, s_3, s_4) \quad (2)$$

$$F = Q(f_1, f_2, f_3, f_4) \quad (3)$$

Here

$$P = \frac{s_1 + s_3}{s_2 + s_4} \quad (4) \text{ And}$$

$$Q = f_1 + f_2 + f_3 + f_4 \quad (5)$$

Form (1) (2) (3) (4) (5)

$$K = (S - F) + E \quad (6)$$

Using (6) we will get the direct trust value calculated by x on y.

G. Obtaining indirect trust

Indirect trust means getting the trust value of required node form its neighbours. Indirect trust value of x on node y is denoted as idt_{xy}. This will follow the below steps. Here node x requesting the trust value of y form a node n. where n belongs to set of nodes N which in range of x and y such that $t_{xn} \geq tt$, tt is threshold trust.

Algorithm: Obtaining indirect trust on y form n

[a] Node x sends IDTREQ to node(s) n □ N.
[b] If node n has direct trust value on y, then it will reply back with IDTREP.
[c] If n does not have direct trust value record it will forward the IDTREQ to its one hop neighbours.
[d] A max-hop and TTL field values are maintained in IDTREQ to limit the request.

Like this we will obtain the indirect trust or recommended trust. After this while deciding the indirect trust value, the concept of fuzzy logic is used.

H. Aggregator Functionality

The aggregator (AG) uses dt_{xy} and idt_{xy} to calculate the total trust value of x on y. For this it use $tt_{xy} = W_1 dt_{xy} + W_2 idt_{xy}$ ---- (7) where tt_{xy} is the total trust value between x and y. then we will use the fuzzy logic to take decision on that node. To take decision we have to calculate the trust evaluation grade *tg* using the below equation.

$$tg(tt) = \begin{cases} 1 \ When\ tt \geq tht \\ \frac{1}{1+(tt-tht)^2} \ when\ tt \in (0, tht] \end{cases} \quad (7)$$

Where tg is trust evaluation grade function, tt is evaluated total trust and tht is the threshold trust of node. The outcome of the above function lies between [0, 1]. Depending on this *tg* value we will assign the duties to the nodes.

I. Energy Measurement

Energy used at each node is calculated using [10]. To calculate the energy used by a node the initial node configuration is necessary. In node configuration initial energy, ideal power, sense power details should be

specified. The total node energy expenditure at a node due to another node in the network can be calculated as follows.

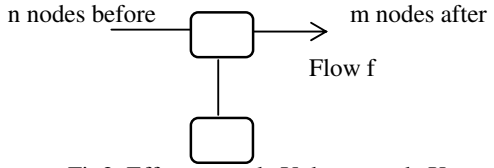

Fig2. Effect on node Y due to node X

$$E_{\frac{Y}{X}} = p_{n>0}\left(p_{X\to Y} E_{T_{ack}} + p_{X\to Y} E_{R_{ack}}\right) + p_{m>0}\left(p_{X\to Y} E_{T_{pck}} + p_{X\to Y} E_{R_{pck}}\right) \quad (8)$$

Where

$E_{\frac{Y}{X}}$ = Energy spent at node Y due to node X

E_{Tack}= Energy spent for one acknowledgement

E_{Tpck}= Energy spent for transmission of one data packet

E_{Rack}= Energy Spent for reception of one ACK packet

E_{Rpck}= Energy spent for reception of one data packet

$$P_q = \begin{cases} 1 \; if \; q \; is \; true \\ 0 \; otherwise \end{cases}$$

This model simplifies packet exchange by including data and the ACK packets only.

J. Disseminator

The nodes in MANET are move dynamically. The disseminator uses this mobility of the nodes for disseminating trust value. Whenever a new trust value is calculated for a node, it is initially propagated to a subset of nodes that are at the nearest distance (1-hop) in the network. This subset of the neighbouring nodes is represented as '*forwarded'*. At regular intervals, the neighbouring nodes in the network participate in dynamic exchange of certificates. The number of elements in the subset *'forwarded'* determines the effective convergence time of trust information among nodes that are currently and in near future would be the neighbours of node. Flooding mechanism is used to supply the calculated trust value of a node to all its neighbours. The number of hops required found flooded can be determined dynamically by making neighbours of the node send neighbourhood information along with observed behaviour of the node. The certificates are piggybacked on routing packets, and thus involve no communication overhead.

V. FUTURE ENHANCEMENTS

In future we can add some additional watchdog mechanisms for supervisor module. By considering some additional factors like wrong routing, replay packets generated, battery exhaustion, link broken will add more accuracy for the calculation of trust value. Fuzzy logic's additional enhancement gives more benefits to find the selfish nodes as a result we will get trusted/ secure communication in mobile ad hoc network. By considering the more reasons for packet dropping we will get more accurate trusted network. For getting more secure network we should consider the more security factors.

VI. CONCLUSION

In this paper discusses the solution to calculate the trust in mobile ad hoc network and to identify the selfish nodes taking energy utilization factor as an additional factor in calculating direct trust. In this paper a fuzzy based approach to evaluate the accurate final trust of the target node.

VII. REFERENCE

[1] Nikos Komninos, Dimitris, Christos Douligeris Detecting unauthorised and compromised nodes in mobile ad hoc networks in: Elsevier ScienceDirect 2005.

[2]. Rajan Shankaran, Vijay Varadharajan, Mehmet A. Orgun, and Michael Hitchens Critical Issues in Trust Management for Mobile Ad-Hoc Networks in: IEEE IRI 2009, July 10-12, 2009, Las Vegas, Nevada, USA.

[3] Jaydip Sen A Distributed trust management framework for detecting malicious packet dropping nodes in a mobile ad hoc network; International Journal of Network Security & Its Applications (IJNSA), Vol.2, No.4, October 2010

[4] Junhai Luo, Xue Liu, Mingyu Fan A trust model based on fuzzy recommendation for mobile ad-hoc networks in: ELSEVIER Computer Nteworks 2009.

[5] Zhiwei Qin, Zhiping Jia, Xihui Chen Fuzzy Dynamic Programming based Trusted Routing Decision in Mobile Ad Hoc Networks in: Fifth IEEE International Symposium on Embedded Computing 2008.

[6] Pallavi Khatri, S.Tapaswi, U.P.Verma Fuzzy Based Trust Management for Wireless Ad hoc Networks in: Int'l Conf. on Computer & Communication Technology 2010.

[7] Sonja Buchegger, C´edric Tissi`eres, and Jean-Yves Le Boudec A Test-Bed for Misbehavior Detection in Mobile Ad-hoc Networks — How Much Can Watchdogs Really Do? Proceedings of the Sixth IEEE Workshop on Mobile Computing Systems and Applications (WMCSA2004).

[8] M.Virendra, M.jadliwala et al, Quantifying Trust in Mobile Ad Hoc Networks, IEEE international conference on integration of knowledge intensive Multi-Agent Systems, pp.65-70. 2005

[9] A.A.Pirzada, A.Dutta, Propagating trust in Ad Hoc networks for Reliable routing. IEEE International workshop on wireless Ad Hoc networks pp 58-62, 2004

[10] Geraud Allard1, Pascale Minet1, Dang-Quan Nguyen1, and Nirisha Shrestha2 Evaluation of the Energy Consumption in MANET.