International Journal of Engineering Applied Sciences and Technology, 2016
Vol. 2, Issue 1, ISSN No. 2455-2143, Pages 121-127
Published Online November-December 2016 in IJEAST (http://www.ijeast.com)

AN ON-DEMAND TRUST BASED SECURE ROUTING PROTOCOL IN MOBILE AD HOC NETWORKS

G. Santhi,
Assistant Professor,
Department of Information Technology,
Pondicherry Engineering College, India.

Abstract: **A Mobile Ad-hoc Network (MANET) allows wireless nodes to form a network without requiring a fixed infrastructure. Due to dynamic nature of MANET the malicious node can easily enter into the network and disrupts the data transmission. In order to reduce the hazards from such nodes and enhance the security of network, it is important to rate the trustworthiness of other nodes without relying a central authority to build up a 'trust' environment. In this research work an Enhanced Trust-based Secured Source Routing protocol (ETSR) is proposed. In which each node predicts their neighbors' future behaviors and selects the shortest trusted route to transmit required packets. Experiments have been conducted to evaluate the efficiency and effectiveness of the proposed ETSR which shows better performance in malicious node identification and attack resistance.**

Keywords -- **Security, Trust, Fuzzy logic, malicious nodes, Mobile nodes**

I. INTRODUCTION

A mobile ad hoc network (MANET) is a system of wireless mobile nodes that can freely and dynamically self-organize in arbitrary and temporary network topologies without the need of a wired backbone or a centralized administration. People and devices can be seamlessly internetworked in areas without any pre-existing communication infrastructure. A MANET can be used to provide access to crisis management applications, such as in a disaster recovery, where the entire communication infrastructure is destroyed and establishing communication quickly is crucial.

Due to the distributed nature, openness in network topology and absence of a centralized administration in the management, MANETs often suffer from attacks by malicious nodes [1]. 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. With authentication and encryption mechanisms, secure routing protocols have been developed to ensure properties such as confidentiality and integrity. These protocols require a centralized trusted third party, which is impractical for MANETs [3]. Moreover, the traditional cryptosystem based security mechanism is typically used to resist the external attacks. They show inefficiency in handling the attacks from the internal malicious nodes which may lead to serious influence on the security, the confidentiality, and the life cycle of the whole network.

Recently, various research works on building up 'trust' among distributed network nodes to simulate cooperation and improving the performance and security of the network. Liu et al. proposed a trust model for mobile ad hoc net- works which uses both cryptography and trust [5]. In this model, each node is initially assigned a trust level. The concepts discussed in this paper are generic and do not rely on centralized control, key distribution protocols, or any particular routing protocol

Sun et al. [6, 7], proposed a model based on entropy. They introduced an entropy function to represent the trust value between two nodes, which captured the dynamic nature of trust evidence. To compute the indirect trust value, Sun's models used trust value iteration techniques considering multi-level directed graph. When more nodes involved, the convergence speed of this method is exponentially slow, and its scalability becomes an issue.

In the opinion of Pirzada and McDonald [8,9], the reliance on a central entity is against the very nature of mobile ad hoc networks, which are supposed to be

International Journal of Engineering Applied Sciences and Technology, 2016
Vol. 2, Issue 1, ISSN No. 2455-2143, Pages 121-127
Published Online November-December 2016 in IJEAST (http://www.ijeast.com)

improvised and spontaneous. They presented a trust-based model for communication in pure mobile 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 this network.

Even though all these protocols perform well they suffer with routing cache problem, which leads to network overhead. Routing cache stores the recent routes that have been used for data transmission for future use which leads to more control overhead and reduces the throughput. To overcome these problems, in this research work an Enhanced Trust-based Secured Source Routing protocol (ETSR) is proposed. This Adaptive Trust Level Classification protocol provides the required security without degrading the performance of the system and with no overheads to the system. By way of considering the average trust values of the intermediate nodes to compute the route's trust, our proposed work overcomes the problem of the routing cache.

II. DESIGN OF ENHANCED TRUST BASED SECURE ROUTING PROTOCOL

Figure 1 describes the system architecture of Enhanced Trust based Secure Routing protocol.

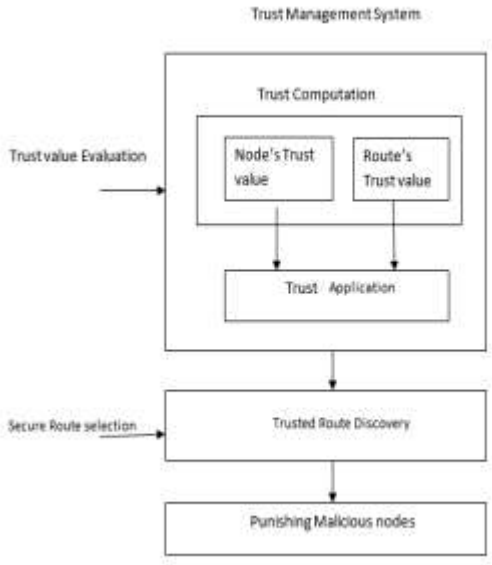


Figure 1 System architecture of ETSR

The trust computation block involves the process of computation of the node's trust value based on the adopted trust model. The node's trusts such as the node's historical trust values and the node's current trust values are computed in this block. The node's historical trust value is computed by observing the node's behavior. Based on the historical trust value computed and the node's capability level, the node's current trust is computed by using the fuzzy logic rules prediction. These two trust values are of great importance and play a major role in establishing a trusted route to the destination.

A. Trust Computation

In ad hoc networks, 'trust' is a relationship between two neighbor entities and it is defined as the reliability, timeliness, and integrity of message delivery to their intended next-hop [10]. In our model, there are three types of trust, which are historical trust, current trust and route trust is considered.

(i) Node's historical trust: It is estimated by the node's physical neighbours based on historical interaction information [11]. In our proposed work, the packet forwarding ratio is used as the single observable factor for assessing this trust. Two trust factors, which are control packet forwarding ratio (CFR) and data packet forwarding ratio (DFR), are assigned weights in order to determine the overall historical trust of an evaluated (or monitored) node.

Forwarding Ratio (FR): It is the proportion of the number of packets forwarded correctly to the number of those supposed to be forwarded. Correct forwarding means a forwarding node not only transmits a packet to its next hop node but also forwards devotedly. At time t, FR(t) is computed as follows:

$$FR(t) = \frac{N_{cor}(t)}{N_{all}(t)} \cdots\cdots\cdots\cdots (1)$$

where,

$N_{cor}(t)$, represents the cumulative count of correct forwarding packets and

$N_{all}(t)$,signifies the total count of all requesting packets from time 0 to t.

Two trust factors (CFR and DFR) are assigned weights in order to determine a monitored node's trust. At time t, the historical trust value of a node TV_{ij} is calculated as:

$$TV_{ij}(t) = w1 \times CFR_{ij}(t) + w2 \times DFR_{ij}(t) \cdots (2)$$

International Journal of Engineering Applied Sciences and Technology, 2016
Vol. 2, Issue 1, ISSN No. 2455-2143, Pages 121-127
Published Online November-December 2016 in IJEAST (http://www.ijeast.com)

where,

$CFR_{ij}(t)$ and $DFR_{ij}(t)$ represent control packet forwarding ratio and data packet forwarding ratio respectively observed by node v_i for forwarding node v_j, and the weights *w1* and *w2* (*w1, w2*> 0 and *w1* + *w2* = 1) are assigned to $CFR_{ij}(t)$and $DFR_{ij}(t)$respectively at time t.

node ID	Node's historical trust value	N_{cor} and N_{all} for control packets	N_{cor} and N_{all} for data packets	Packet buffer

Figure 2 Data structure of a node with Trust value

(ii) Node's current trust: A node's current (or prediction) trust predicts this evaluated node's future behaviours for the next time moment [12]. In our model, it is computed from the node's historical trust based on the fuzzy logic rules prediction method. In our proposed work, at time t, we use the term 'trust value' TV(t) for a node's current trust value, for simplicity of representation.

let C(t) represents for the node's capability level on providing packets transmission services at time t, which includes the remnant utilization ratio of battery, local memory, CPU cycle, and bandwidth at that point; let TV(t + 1) refers to the node's trust level at time t + 1.

Table 1. Logical rules prediction on trust levels

C(t)	TV(t)			
	VL	L	M	H
VL	VL	VL	VL	VL
L	VL	VL	L	M
M	VL	L	M	H
H	VL	L	M	H

Assume the fuzzy membership function of TV(t) or TV(t + 1) consists of four fuzzy sets: VeryLow (VL-malicious node), Low(L-low trust worthy node), Medial(M-trustworthy node) and High (H-complete trustworthy node), and the fuzzy member function of C(t) also consists of four fuzzy sets: VeryLow (VL-cannot afford to provide services), Low (L-low capability level), Medial(M-medium capability level) and High(H-high capability level), respectively. Combined with social control theory, we give the fuzzy inference rules as follows

The rules in the above table actually establish a mapping function from TV(t)×C(t) to TV(t + 1), which is based on the analysis of the node's historical behaviors and current conditions.

Corresponding with each rule, there is an inference relationship R_l:

$$R_l = TV_t \times C_t \times TV_{t+1} \cdots\cdots\cdots\cdots (3)$$

That is for ∀h∈TV(t),C ∈ C(t), u ∈ TV(t + 1), we have

$$R_l(h, c, u) = TV(h)^{C(c)^{TV(u)}} \cdots\cdots\cdots (4)$$

For all the n rules we have the fuzzy inference relationship

$$R_l(h, c, u) = \bigvee_{l=1}^{n} R_l(h, c, u) \cdots\cdots\cdots\cdots (5)$$

For each pair of given $(TV(t)^*, C(t)^*)$, using the general total relationship R, we can obtain an output:

$$TV(t+1)^* = (TV(t)^* \times C(c)^*) \circ R \cdots\cdots\cdots\cdots (6)$$

Then with the help of the maximum membership degree approach, we can get an explicitly node's current trust u* ∈ [0, 1] by defuzzification. We can recycle the method to update this node's trust. Finally, each node additionally owns a trust table with items defined as follows:

Node_ID	Neighbour_ID	TV	Black-List

Figure 3 Structure of Node's Trust Table

Node_IDis the identification (ID) of node v_i; Neighbour_ID is the identification (ID) of node v_i's neighbour;TV is the trust value that node v_i has about any neighbor; Black-list indicates whether node v_i consider this concerned neighbor node (e.g., monitored node v_j) as a malicious node or not.

(iii) Trust Application

A node's trust value not only provides a relative identification between the normal node and the malicious node, but also offers a prediction of this node's future behaviours. We present a simple case to illustrate the trust application. The service node s determines route trust requirements basing on the service level. We can simply divide the service in a

International Journal of Engineering Applied Sciences and Technology, 2016
Vol. 2, Issue 1, ISSN No. 2455-2143, Pages 121-127
Published Online November-December 2016 in IJEAST (http://www.ijeast.com)

file sharing system into three levels: important documents sharing (I), the less important documents sharing (L), and the regular documents sharing (R). For those shared files, we define a mapping function f (as shown in Eq. (8)).

$$f(RouteTV_{sd}) = \begin{cases} I\ 0.9 \le RouteTV_{sd} \le 1 \\ L\ 0.8 \le RouteTV_{sd} \le 0.9 \\ R\ 0.7 \le RouteTV_{sd} \le 0.8 \end{cases} \cdots\cdots \cdots\cdots (8)$$

We can also set different boundary values (e.g., $0.95 \le RouteTV_{sd} \le 1$) corresponding with the special security needs of the network. According to this file sharing system, we define a simple grading criteria for node's trust levels, which is shown in Table 2

Table 2 Trust levels of nodes

Trust Level	Trust Value	Node's state
1	[0,η)	Malicious Node
2	[η,0.7)	Suspicious Node
3	(0.7,0.8)	Low Trustworthy Nodes
4	[0.8,0.9)	Trustworthy Nodes
5	[0.9,1]	Complete Trustworthy Nodes

(iv) Computation of Route's Trust

The route's trust value is calculated as the average trust values of the intermediate nodes between the source and the destination. It is denoted by $RouteTV_{sd}(t)$. It is calculated as:

$$RouteTV_{sd}(t) = \frac{\sum\{TV_{ij}(t) | v_i, v_j \in P \text{ and } v_i \to v_j\}}{n} \cdots\cdots (7)$$

where,

$TV_{ij}(t)$ – represents the Node's trust value,
v_i and v_j are any two adjacent nodes along the route P,

In the example below, the Route Trust for the path (A→B→D→F) is calculated as 0.93 by using the equation (7).

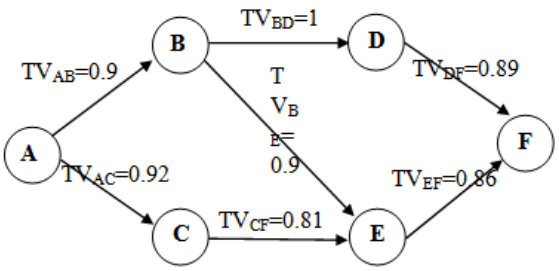


Figure 4 Computation of Route Trust

Punishing malicious nodes

Punishing malicious nodes for a specific time is a solution for the problem of dynamic modification of a node's behaviour. Every node in the black-list has a specific time (termed as the isolated time) in which the evaluated node v_i is regarded as a malicious node by the owner (evaluating node v_j) of the black-list. During the isolated time, node v_i is insulated from forwarding packets. After the time, node v_i will be removed from the black-list and its trust will be set to the black-list trust threshold.

B. Trusted Route Discovery

The procedure of trust-based secured source routing protocol is given in the flow chart.

(i) Initially the source node that needs to transmit the data to destination node, initiates the routing process. The source node checks for the unexpired qualified node in this routing cache to the destination node. If there exists an unexpired qualified path to the destination node, then the source node starts transmitting the data through that path. If there exists no such unexpired qualified path to the destination, the node sends the FLOW-REQ message containing the source and the destination node IDs to all its one-hop neighbours.

(ii) The neighbouring node on receiving the FLOW-REQ message, checks the destination node's ID with its own ID. If the ID of the neighbouring node does not match with the destination node, it forwards the FLOW-REQ message to all its one-hop neighbours by appending its own ID and Trust value to the FLOW-REQ message.

International Journal of Engineering Applied Sciences and Technology, 2016
Vol. 2, Issue 1, ISSN No. 2455-2143, Pages 121-127
Published Online November-December 2016 in IJEAST (http://www.ijeast.com)

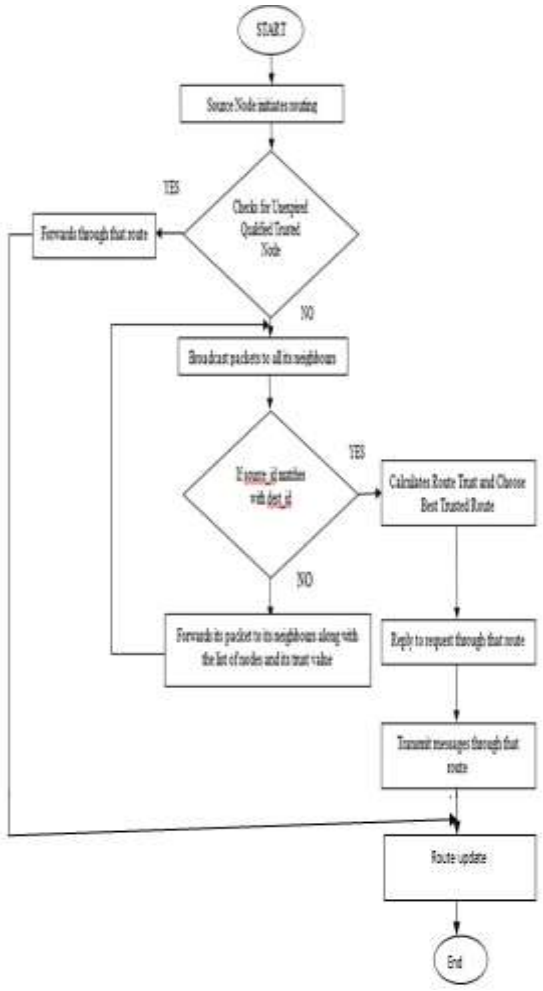


Figure 5 Flow chart of the Trust-based Secured Source Routing protocol

(iii) If the ID of the neighbouring node matches with the ID of the destination node, it sends a reply (FLOW-SETUP message) to source node through the best qualified trusted path.

(iv)The source node on receiving the FLOW-SETUP message from the destination node, it transmits the message through that path. The source node also makes a route update for a regular period of time

III. RESULTS AND DISCUSSIONS

The proposed ETSR protocol work has been simulated in various network scenarios using the C++ programming language. A discrete event simulation is done to test the operation effectiveness of the proposed work. In this section we describe the simulation model and the simulation procedure.

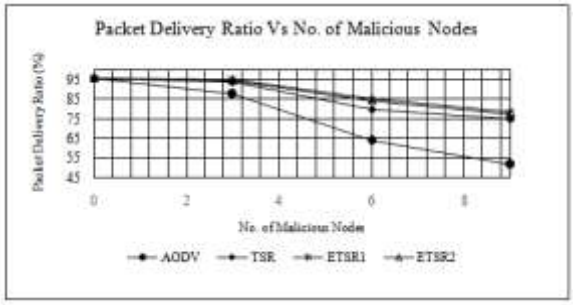
Table 3 Simulation Parameters

Parameter	Value
MAC	MAC/802.11
Transmission Range	250 m
Traffic Flow	CBR(Constant Bit Rate)
Packet Size	512 bytes
No. of nodes	30
Simulation Area	1000 m × 1000 m
Node mobility Speed	0 – 30 m/s
Mobility Pattern	Random Way Point
Simulation Time	100 s

The performance of proposed ETSR is compared with TSR, and AODV against network throughput, packet delivery ratio, end-to-end delay and control overhead by varying the number of malicious nodes from 0 to 9 and by fixing the maximum speed as 10m/s and the blacklist trust threshold as 0.5.

a. Packet delivery ratio

With no malicious nodes the packet loss rate is about 4%.The delivery ratio of ADOV declines sharply, while TSR, ETSR1 and ETSR2 degrade gently as the number of malicious nodes increases. The delivery ratio of TSR, ETSR1 and ETSR2 are always higher than AODV, this is because using the trust, TSR, ETSR1 and ETSR2 allows no malicious nodes to forward packets. The delivery ratio of ETSR2 drops from 96% to 78% as the number of malicious nodes increases from 0 to 9. The reason for this drop is that with the proportion of malicious nodes increases, the probability of suspect or low trustworthy nodes existed on the routing route also increases, leading to descend the packet delivery ratio. From the sharp attenuation in AODV, we find that, malicious nodes make huge damage to the whole network, and more malicious nodes are, the more serious their damage is.



International Journal of Engineering Applied Sciences and Technology, 2016
Vol. 2, Issue 1, ISSN No. 2455-2143, Pages 121-127
Published Online November-December 2016 in IJEAST (http://www.ijeast.com)

Figure 6 Packet Delivery Ratio Vs Number of malicious nodes

b. Average end-to-end latency

The average end-to-end latency in TSR, ETSR1 and ETSR2 increases slowly as the number of malicious nodes increases. This average latency is mainly caused by queuing and retransmission delays. This reason is that, the TSR and ETSR add trust concept, along with the malicious nodes increase, the routing route established by these methods may add hops, which results in the greater delay. However, the average latency in AODV ascends sharply and there is an obvious reduction in the average latency with TSR, ETSR1 and ETSR2 compared to AODV. There are two reasons: (1) in the process of route discovery and Path Selection, the network can avoid malicious nodes; (2) the availability of alternative routes eliminates delay caused by route rediscoveries in AODV, while multiple candidate mechanisms avoid route rediscoveries in TSR and ETSR, which contribute to effectively reduce the end-to-end latency.

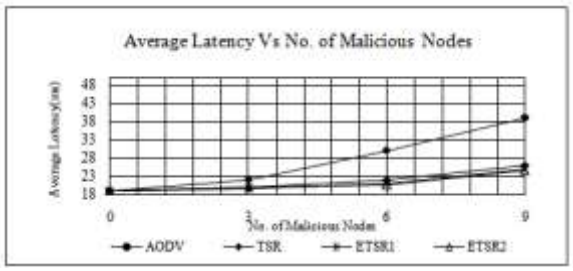


Figure 7 Average end-to-end latency Vs Number of malicious nodes

c. Routing Packet Overhead

When the number of malicious nodes increases to 9(30% of whole nodes), the routing packet overhead of AODV, TSR, ETSR and ETSR2 are approximately 0.47, 0.26, 0.25 and 0.24 respectively as shown in the Figure 6.7. The routing packet overhead of ETSR is smaller than TSR and AODV. When the number of malicious nodes is smaller than 4, the routing packet overhead in TSR and ETSR is bigger than in AODV, the reason is that, the increased control packets in TSR and ETSR are primarily due to their route discovery mechanism that broadcasts more Flow-REQ and Flow-SETUP packets to look for trustworthy routes to destinations. However, when the number of malicious nodes is bigger than 5, the routing packet overhead in TSR and in ETSR is smaller than AODV, because of that the huge damage on routing path from malicious nodes. In AODV, due to the absence of the participation of the trust model, along with the increase number of malicious nodes, almost all of the routing route has the participation of malicious nodes which launch a constant 30% probability of modification attack, leading to the sharply increase in routing packet overhead.

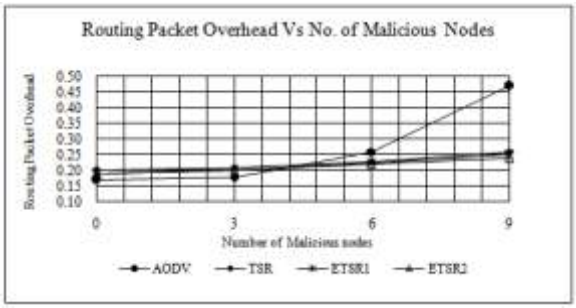


Figure 8 Routing Packet overhead Vs Number of malicious nodes

d. Network throughput

Figure 9 shows that our proposed approach can get an obvious throughput than TSR and AODV. Corresponding with Figure 6.5 lower the delivery ratio lower is the network throughput.

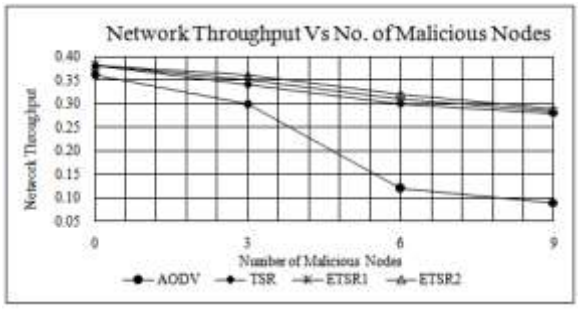


Figure 9 Network throughput Vs Number of malicious nodes

The experiment results in tests 1 and 2 show that ETSR performs better than TSR and AODV. With the help of 'trust', we use smaller increment of routing overhead to exchange with the bigger enhancement of the network security and performance, as ETSR gives higher delivery ratio, network throughput and detection ratio for malicious nodes. Overall, ETSR2 shows a better performance than ETSR1. That proves our observation that control

packets play a more important role than data packets in a MANET. For a real-world application, the factor values can be adjusted to satisfy the need of the trustworthiness and performance, depending on the requirement, characteristics, and environment of the application.

IV. CONCLUSION

By taking the node's prediction trust value as the input, a novel reactive trusted routing protocol extending from the standard Source Routing Mechanism called Enhanced Trust-based Secured Source Routing protocol (ETSR) is proposed in this work. It can improve the TSR to kick out the untrustworthy nodes such that a reliable passage delivery route is obtained and alleviate the attacks from malicious nodes. The proposed ETSR provides a flexible and feasible approach to choose a better route in all path candidates with trust constraint. Performance comparison of these routing protocols (AODV, TSR and ETSR) shows that ETSR is able to achieve a remarkable improvement in the packet delivery ratio, network throughput and defend some classical malicious attacks (e.g., fractions of modification, grayhole and blackhole attacks). For future work, other criterion can be used to determine the optimum route to set up the flow such as route Quality of Service (QoS), load and delay.

V. REFERENCES

[1] X. Li, Z. Jia, P. Zhang, R. Zhang, H. Wang, Trust-based on-demand multi path routing in mobile ad hoc networks', IET Special Issue onMulti-Agent & Distributed Information Security 4 (4) (2010) 212–223.

[2] E.M. Royer, C.K. Toh, A review of current routing protocols for ad hocmobile wireless networks, IEEE Personal Comminations Magazine 6(2) (1999) 46–55.

[3] D. Johnson, D. Maltz, Dynamic source routing in ad hoc wireless networks, in: I. Tomasz, K. Hank (Eds.), Mobile Computing, first ed.,Kluwer Academic Press, 1996, pp. 153–181.

[4] C.E. Perkins, E.M. Royer, S.R. Das, Ad-hoc on-demand distance vector routing, in: Proceedings of International Workshop on Mobile Computing Systems and Applications (WMCSA), New Orleans, USA, February 1999, pp. 90-100.

[5] J. Lundberg, Routing Security in Ad hoc Networks, Technical ReportTik110.501, Helsinki University of Technology, 2000.

[6] W.L.H. Deng, D.P. Agrawal, Routing security in wireless ad hocnetworks, IEEE Communications Magazine (2002) 70–75.

[7] N. Griffiths, A. Jhumka, A. Dawson, R. Myers, A simple trust model foron-demand routing in mobile ad-hoc networks, in: Proceedings ofInternational Symposium on Intelligent Distributed Computing (IDC2008), 2008, pp. 105–114.

[8] S. Marti, T.J. Giuli, K. Lai, M. Baker, Mitigating routing misbehavior inmobile ad hoc networks, Mobile Computing and Networking (2000), 255–265.

[9] T. Hughes, J. Denny, P.A. Muckelbauer, J. Etzl, Dynamic trust appliedto ad hoc network resources, in: Proceedings of the AutonomousAgents and Multi-Agent Systems Conference, 2003, pp. 273–280.

[10] K. Meka, M. Virendra, S. Upadhyaya, Trust based routing decisions inmobile ad-hoc networks, in: Proceedings of the Workshop on SecureKnowledge Management (SKM 2006), 2006.

[11] C.D. Jensen, P.O. Connell, Trust-based route selection in dynamicsource routing, Proceedings of International Conference on TrustManagement (2006) 150–163.

[12] A.A. Pirzada, C. McDonald, A. Datta, Performance comparison oftrust-based reactive routing protocols, IEEE Transactions on MobileComputing 5 (6) (2006) 695–710.