# Secure Trust Based Dynamic Source Routing in MANETs

Yogendra Kumar Jain, Nikesh Kumar Sharma

**Abstract**—Secure routing is an important issue in MANETs. A particularly devastating attack in wireless networks is the black hole attack. The performance of ad hoc networks depends on cooperation and trust among distributed nodes. To enhance security in ad hoc networks, it is important to evaluate trustworthiness of other nodes without centralized authorities. As a result, an efficient algorithm to detect black hole attack is important. In this paper, to improve the quality a modified design of trust based dynamic source routing protocol is proposed. Each node would evaluate its own trusted parameters about neighbors through evaluation of experience, knowledge and recommendations. This protocol discovers multiple loop-free paths which are evaluated by hop count and trust. This judgment provides a flexible and feasible approach to choose a shortest path in all trusted path. The proposed protocol reduces the packet loss due to malicious nodes to a considerable extent thereby enhancing the performance. We also compare the simulation results of with and without the proposed secure trust based model. The simulation results demonstrate that the PDR for STBDSR falls from 92% to 80%.

**Keywords**—MANET, DSR, Security, Black hole, Multi-Party Computation, Trusted Path, Trust Model

## 1 INTRODUCTION

Mobile Ad Hoc Network (MANET) [1] is one kind of new wireless network structures. Unlike devices in traditional Wireless LAN solutions, all nodes are movable and the topology of the network is changing dynamically in an Ad Hoc Networks, which brings great challenges to the security of Ad Hoc Networks. A wide variety of security attacks such as black hole and grey-hole attacks address the routing procedure. In the black hole and grey-hole attacks the selfish nodes are refused to forward all or part the traffic received from its neighbors. Security and robustness of the protocol would be improved if nodes could make informed decisions regarding route selection based on transmitted route requests and additional information contained in received route replies. Thus, additional information on the nature of routes would enable the Source node to choose a route that best serves its purpose.

We present a flexible trust model based on the concept of human trust and apply this model to ad hoc networks. Our model builds, for each node, a trust relationship to all neighbors. The trust is based on previous individual experiences of the node, knowledge and on the recommendations of its neighbors. The recommendations improve the trust evaluation process for nodes that do not succeed in observing their neighbors due to resource constraints or link outages. The ability of assessing the trust level of its neighbors brings several advantages. First, a node can detect and isolate malicious behaviors, avoiding relaying packets to malicious neighbors. Secondly, cooperation is stimulated by selecting the neighbors with higher trust levels. Nodes learn based on the information exchanged with trustworthy neighbors to build a knowledge plane [2].

In our model nodes interact only with its neighbors. As a result, nodes do not keep trust information about every node in the network. Keeping neighborhood information implies significant lower energy consumption, less processing for trust level calculation, and less memory space. It also fits well to ad hoc networks, which are usually composed of portable devices with power, processing, and memory restrictions [3]. Moreover, topology changes, due to mobility or battery constraints, make it difficult to maintain information for all nodes [4]. Another result is that recommendations are only exchanged between neighbors, that is, recommendations are not forwarded. This approach also minimizes the probability of false recommendations since the number of received recommendations is significantly smaller and there is no intermediate node to increase the uncertainty of the information. Besides, a node can always balance the received recommendations with its own experiences to calculate the trust level because nodes do not calculate the trust level of nodes that are not neighbors. The decrease in the number of messages sent not only alleviates the network traffic, but also decreases the energy consumption.

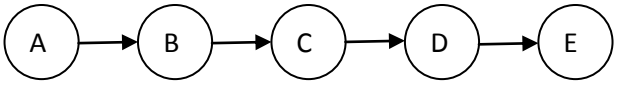
- *Dr. Yogendra Kumar Jain presently working as head of the department Computer Science & Engineering at Samrat Ashok Technological Institute Vidisha M.P. India. PH-+91-7592-250408. E-mail: ykjain_p@yahoo.co.in*
- *Mr. Nikesh Kumar Sharma presently pursuing M.Tech from the department, Computer Science & Engineering at Samrat Ashok Technological Institute, Vidisha M.P. India. Mobile-7489260305. Email:nikesh_sharma0084@yahoo.com*

The paper is organized as follows. In the Section 2, we discuss DSR, black hole and trusted path. We expose the related works in Section 3. We present our model in Section 4. Section 5 shows our simulation results. In Section 6 we present our conclusions.

## 2 LITERATURE SURVEY

### 2.1 Over View of DSR Protocol

DSR is a source routing in which the source node starts and take charge of computing the routes [5]. At the time when a node S wants to send messages to node T, it firstly broadcasts a route request (RREQ) which contains the destination and source nodes' identities. Each intermediate node that receives RREQ will add its identity and rebroadcast it until RREQ reaches a node n who knows a route to T or the node T. Then a reply (RREP) will be generated and sent back along the reverse path until S receives RREP. When S sends data packets, it adds the path to the packets' headers and starts forwarding. During route maintenance, S detects the link failures along the path. If it happens, it repairs the broken links. Otherwise, when the source route is completely broken, S will restart a new discovery.




1. A----->B : (A) ID=2

2. B---- > C: (A, B) ID=2

3. C---->D :( A, B, C) ID=2

4. D----> E :(A, B, C, D) ID=2

Fig. 1. Route Discovery Process

To initiate the Route Discovery [6], the source transmits a ROUTE REQUEST (RREQ) message as a single local Broadcast packet, which is received by (approximately) all nodes currently within wireless transmission range of source. Each RREQ message identifies the initiator and target of the Route Discovery, and also contains a *unique request id,* determined by the initiator of the REQUEST. Each RREQ also contains a record listing the address of each intermediate node through which this particular copy of the RREQ message has been forwarded. This route record is initialized to an empty list by the initiator of the Route Discovery. When another node receives a RREQ, if it is the target of the Route Discovery, it returns a ROUTE REPLY (RREP) message to the initiator of the Route Discovery, giving a copy of the accumulated route record from the RREQ; when the initiator receives this ROUTE REPLY, it caches this route in its Route Cache for use in sending subsequent packets to this destination. Otherwise, if this node receiving the RREQ has recently seen another RREP message from this initiator bearing this same request id, or if it finds that its own address is already listed in the route record in the RREQ message, it discards the REQUEST. Otherwise, this node appends its own address to the route record in the ROUTE REQUEST message and propagates it by transmitting it as a local broadcast packet with the same request id.

Route Maintenance [6] is the mechanism by which source node is able to detect, while using a source route to destination node, if the network topology has changed such that it can no longer use its route to destination node because a link along the route no longer works. When Route Maintenance indicates a source route is broken, source node can attempt to use any other route it happens to know to destination node, or can invoke Route Discovery again to find a new route. Route Discovery and Route Maintenance each operate entirely *on demand*. For example, DSR does not use any periodic routing advertisement, link status sensing, or neighbor detection packets, and does not rely on these functions from any underlying protocols in the network. This entirely on-demand behavior and lack of periodic activity allows the number of overhead packets caused by DSR to scale all the way down to zero, when all nodes are approximately stationary with respect to each other and all routes needed for current communication have already been discovered. As nodes begin to move more or as communication patterns change, the routing packet overhead of DSR *automatically* scales to only that needed to track the routes currently in use. In response to a single Route Discovery (as well as through routing information from other packets Overheard), a node may learn and cache multiple routes to any destination. This allows the reaction to routing changes to be much more rapid, since a node with multiple routes to a destination can try another cached route if the one it has been using should fail. This caching of multiple routes also avoids the overhead of needing to perform a new Route Discovery each time a route in use breaks. When originating or forwarding a packet using a source route, each node transmitting the packet is responsible for confirming that the packet has been received by the next hop along the source route; the packet is retransmitted (up to a maximum number of attempts) until this confirmation of receipt is received. For example, in the situation illustrated in Fig. 2, node A has originated a packet for E using a source route through intermediate nodes B, C and D. In this case, node

A is responsible for receipt of the packet at B, node B is responsible for receipt at C, node C is responsible for receipt at D, and node D is responsible for receipt finally at the destination E. This confirmation of receipt in many cases may be provided at no cost to DSR, either as an existing standard part of the MAC protocol in use such as the link-level acknowledgement frame defined by IEEE 802.11 or by a *passive acknowledgement*. If neither of these confirmation mechanisms are available, the node transmitting the packet may set a bit in the packet's header to request a DSR-specific software acknowledgement be returned by the next hop; this software acknowledgement will normally be transmitted directly to the sending node, but if the link between these two nodes is uni-directional, this software acknowledgement may travel over a different, multi-hop path. If the packet is retransmitted by some hop the maximum number of times and no receipt confirmation is received, this node returns a ROUTE ERROR message to the original sender of the packet, identifying the link over which the packet could not be forwarded. For example, in Fig. 2, if C is unable to deliver the packet to the next hop D, then C returns a ROUTE ERROR to A, stating that the link from C to D is currently "broken." Node A then removes this broken link from its cache; any retransmission of the original packet is a function for upper layer protocols such as TCP. For sending such a retransmission or other packets to this same destination E, If A has in its Route Cache another route to E (for example, from additional ROUTE Replies from its earlier Route Discovery, or from having overheard sufficient routing information from other packets), it can send the packet using the new route immediately. Otherwise, it may perform a new Route Discovery for this target.

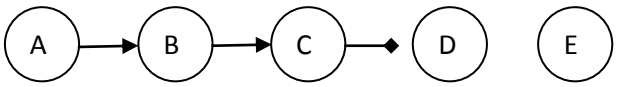



Fig. 2. Route Maintenance Process

The operation of Route Discovery and Route Maintenance in DSR are designed to allow uni-directional links and asymmetric routes to be easily supported. In particular, in wireless networks, it is possible that a link between two nodes may not work equally well in both directions, due to differing antenna or propagation patterns or sources of interference. DSR allows such uni-directional links to be used when necessary, improving overall performance and network connectivity in the system. DSR also supports internetworking between different types of wireless networks, allowing a source route to be composed of hops over a combination of any types of networks available. A node forwarding or overhearing any packet may add the routing information from that packet to its own Route Cache. In particular, the source route used in a data packet, the accumulated route record in a ROUTE REQUEST, or the route being returned in a ROUTE REPLY may all be cached by any node.

### 2.2 Black Hole Attack

In a black hole attack, a malicious node sends fake routing information, claiming that it has an optimum route and causes other good nodes to route data packets through the malicious one. For example, in DSR, the attacker can send a fake RREP (including a fake destination sequence number that is fabricated to be equal or higher than the one contained in the RREQ) to the source node, claiming that it has a sufficiently fresh route to the destination node. This causes the source node to select the route that passes through the attacker. Therefore, all traffic will be routed through the attacker, and therefore, the attacker can misuse or discard the traffic. As for gray hole, its behavior is similar to a black hole. A gray hole does not drop all data packets but just part of packets. We define the *Gray Magnitude* as the percentage of the packets which are maliciously dropped by an attacker. For example, a gray hole is gray magnitude of 60% will drop a data packet with a probability of 60% and a classical black hole has a gray magnitude of 100%. Fig. 3 shows an example of a black hole attack, where attacker A sends a fake RREP to the source node S, claiming that it has a sufficiently fresher route than other nodes. Since the attacker's advertised sequence number is higher than other node's sequence numbers, the source node S will choose the route that passes through node A.

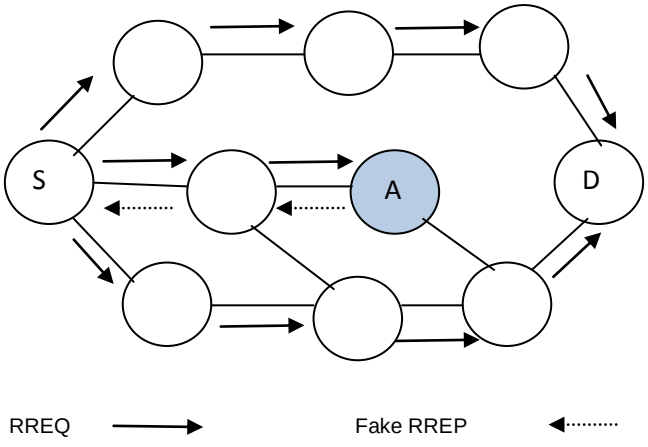



Fig. 3. Example of a Black Hole Attack on DSR.

We evaluate the effects on DSR under varying number of malicious nodes. In the absence of malicious nodes, the typical packet loss is about 1 percent for DSR. As shown in Fig. 4, the packet delivery ratio of protocol degrades sharply as malicious nodes increases. The delivery ratio of DSR drops from 99% to 29% as the number of malicious nodes varies from 0 to 10 and the nodes are moving from 0 to 20m/s. Lower packet delivery ratio means less network throughput. Malicious nodes essentially limit the interactions of nodes in the network.

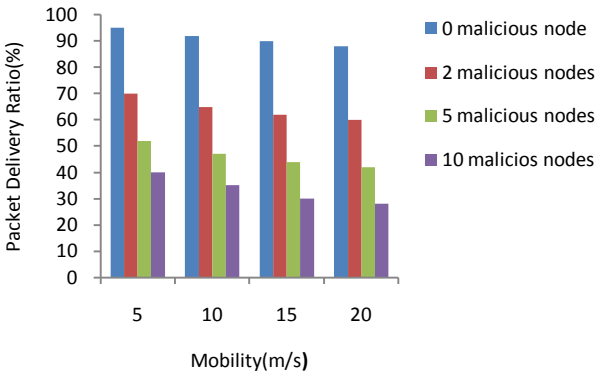



Fig. 4. Packet Delivery Ratio VS Mobility with total 50 nodes

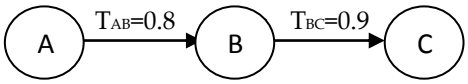
### 2.3 Path's Trust Computation

When a source discovers a path to the destination with the help of forwarding nodes, the trust value of the path is able to be computed through the trust values of nodes among the path [7]. So, in our model the trust of a path $P$ (denoted by $TP(t_i)$) is equal to the minimal one of the nodes' values in the path. i.e.

$$TP(t_i) = min(\{T_{jk}(t_i) \mid n_j, n_k \in P \text{ and } n_j \rightarrow n_k\}) \quad (1)$$

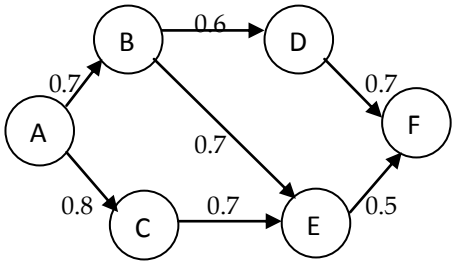
In which, $n_j$ and $n_k$ are any two adjacent nodes among the path $P$ and $n_j \rightarrow n_k$ means that $n_k$ is the next-hop node of $n_j$. The trust computation based on minimal value is similar to opinions in information theory: the information cannot be increased via propagation [8].

As shown in Fig. 5(a), the direction edge from $A$ to $B$ denotes the trust $T_{AB}$. The trust value of path $(A \rightarrow B \rightarrow C)$ is equal to the less one (0.8) of $T_{AB}$ and $T_{BC}$.




(a)

Fig. 5(b) show a complex graph with branches, in which there are three paths from $A$ to $F$ and the path $(A \rightarrow B \rightarrow D \rightarrow F)$ is the most trustworthy path.




(b)

Fig. 5. Path Trust computation

## 3 RELATED WORK

Several works propose monitoring schemes to generate trust values describing the trustworthiness, reliability, or competence of individual nodes. Theodorakopoulos and Baras [9] analyze the issue of evaluating the trust level as a generalization of the shortest-path problem in an oriented graph, where the edges correspond to the opinion that a node has about other node. They consider that nodes use just their own information to establish their opinions. The opinion of each node includes the trust level and its precision. The main goal is to enable nodes to indirectly build trust relationships using exclusively monitored information.

Moe *et al.* [10] proposed a trust-based routing protocol as an extension of DSR based on an incentive mechanism that enforces cooperation among nodes and reduces the benefits that selfish nodes can enjoy (e.g., saving resources by selectively dropping packets). This work is unique in that they used a hidden Markov model (HMM) to quantitatively measure the trustworthiness of nodes. In this work, selfish nodes are benign and selectively drop packets. Performance characteristics of the protocol when malicious nodes perform active attacks such as packet modifications, identity attacks, etc., need to be investigated further.

Sun *et al.* [11] proposed trust modeling and evaluation methods for secure ad hoc routing and malicious node detection. The unique part of their design is to consider trust as a measure of uncertainty that can be calculated using *entropy*. In their definition, trust is a continuous variable, and does not need to be transitive, thus capturing

some of the characteristics of trust in MANETs. However, this work considers packet dropping as the only component of direct observations to evaluate trust.

Balakrishnnan *et al.* [12] developed a trust model to strengthen the security of MANETs and to deal with the issues associated with recommendations. Their model utilizes only trusted routes for communication, and isolates malicious nodes based on the evidence obtained from direct interactions and recommendations. Their protocol is described as robust to the recommender's bias, honest-elicitation, and free-riding. This work uniquely considered a context-dependency characteristic of trust in extending DSR.

Marti *et al.* [13] proposed a reputation-based trust management scheme that consists of a *watchdog* that monitors node behaviors and a *path rater* that collects reputation and takes response actions (e.g., isolating misbehaving nodes as a result of misbehavior detection). This work is an initiative to dynamically incorporate direct observations into trust values for secure routing. It extends DSR (Dynamic Source Routing) but trust evaluation is based only on direct observations

Sen *et al.* [14] proposed a trust-based mechanism to detect malicious packet dropping nodes based on reputation of neighboring nodes, and take into account the decay of trust over time. This work assumes that a pair of public/private keys can be preloaded to prevent identity-related attacks. However, this may not be scalable for a large network.

## 4 PROPOSED TRUST MODEL

The trust model essentially performs the function of trust derivation, computation, and application. In our model, each node derives trust factors from experience, knowledge and recommendation. During trust computation, a linear aggregate method is used to estimate the overall trust in a node according to trust factors, and a minimal value method is used to compute a path's trust.

### 4.1 Trust Level Evaluation

We define the trust level evaluation from node $a$ about node $b$, $Ta(b)$, as a weighted sum of its own trust (monitor), knowledge and the recommendations of neighbors, similar to Virendra *et al.* [15]. The fundamental equation is

$$T_a(b) = W1 \times Q_a(b) + W2 \times K_a(b) + W3 \times R_a(b) \qquad (2)$$

where the variable $Qa(b)$, that ranges from [0,1], represents the trust node $a$ has on node $b$ based on evaluation of experience, $Ka(b)$, that ranges from [0,1], is the knowledge of node $a$ about node $b$ and $Ra(b)$, that ranges from [0,1], is the aggregate value of the recommendations from all other neighbors of node a. The variables $W1$, $W2$ and $W3$ that ranges from [0, 1], and $W1+W2+W3=1$, are parameters in our model that allows nodes to choose the most relevant factor. In our model, the value of $Qa(b)$ is given by

$$Q_a(b) = \beta E_a(b) + (1-\beta) T_a(b) \qquad (3)$$

where $Ea(b)$ is node $a$'s evaluation to node $b$ by directly monitoring packets communication of node $b$, and the variable $Ta(b)$ gives the last trust level value stored in the Trust Table. The variable $\beta$, that ranges from [0, 1], allows different weights for the factors of the equation, selecting which factor is the more relevant at a given moment. In our model, the value of $Ea(b)$ is given by

$$E_a(b) = \frac{P_b^{out}}{P_b^{out} + P_b^{in}} \qquad (4)$$

Where $P_b^{out}$ is the all out-coming packets from node b and $P_b^{in}$ is the all in-coming packets on node b.

### 4.2 Knowledge Computation

$K_a(b)$ is node $a$'s evaluation to node $b$ by directly observing MAC layer link quality between node $a$ and node $b$ on physical layer. This parameter is the probability that the data packet will be successfully transmitted between two network nodes [16]. Computation formula is as follows:

$$K_a(b) = (1 - p_{a,b})(1 - p_{b,a}) \qquad (5)$$

$p_{a,b}$ is packet loss probability from node $a$ to node $b$, while $p_{b,a}$ is packet loss probability from node $b$ to node $a$.

### 4.3 Recommendation Computation

$R_a(b)$ is node $a$'s evaluation to node $b$ by collecting recommendations about node $b$ from neighbors nodes whose trust level is above a certain threshold, to increase the confidence of recommendations. This is given by the equation

$$R_a(b) = \frac{\sum_{i \in \varphi} T_a(i) T_i(b) M_i(b)}{\sum_{i \in \varphi} T_a(i) M_i(b)} \qquad (6)$$

Where $\varphi$ is the group of recommenders. The recommendation of node $i$ about node $b$ is weighted by $Mi(b)$, which defines the maturity of the relationship between nodes $i$ and $b$, measured at node $i$. The relationship maturity is a measure of the time that two nodes have known each other. We use the relationship

maturity to give more relevance to the nodes that know the evaluated neighbor for a longer time. Accordingly, we assume that the trust level of a older neighbor has already converged to a common value within the network and therefore its opinion should be more relevant than the opinion of a new neighbor. It is important to notice that maturity is only considered between the recommender, node *i*, and the node that is being evaluated, node *b*. Malicious nodes can implement an attack exploiting the concept of relationship maturity by attributing fake trust levels. In order to minimize this effect, each node defines a maximum relationship maturity value *Mmax*, which represents an upper bound for the relationship maturity. This value is based on the average maturity relationship value of its most trusted neighbors.

## 5 SIMULATION AND ANALYSIS

In this section, we show results of simulation of DSR with secure trust based model (STBDSR) and pure DSR. The simulations are done in NS2 simulator (version 2.34). In TABLE 1, we summarize the parameters used in the simulations. We define packet delivery ratio (PDR) as the ratio of data packets successfully arrived at destinations to all data packets delivered from sources. Also we use parameters *W1 = 0.4, W2 = 0.4, W3=0.2 and* $\beta$=0.5. We have evaluated packet delivery ratio for different scenarios. The results are showed in Fig. 6.

TABLE 1

SIMULATION PARAMETERS

| | |
|---|---|
| **Total simulation time** | 300 |
| **Simulation area** | 670m*670m |
| **Total number of nodes** | 50 |
| **Radio range** | 250 m |
| **Maximum speed** | 20 m/s |
| **Pause time** | 10s |
| **Data payload** | 512 bytes |
| **Traffic Type** | CBR |
| **Type of Attack** | black hole |
| **Maximum connection** | 10 |

As shown in Fig. 6, the PDR for pure DSR reduced dramatically from 92% to 51% while there are only 5 malicious nodes. In contrast, with the same malicious nodes, the PDR for STBDSR falls from 92% to 80%. The primary reason for this phenomenon is that secure trust based model has been integrated into routing selection process based on trust evaluations. According to trust value computed, each node can wisely decide whether the previous-hop is a trust enough to accept its packets, and also whether the next hop is trust enough to forwarding packets, therefore make sure packets be transmitted on trusted routers and successfully be delivered.

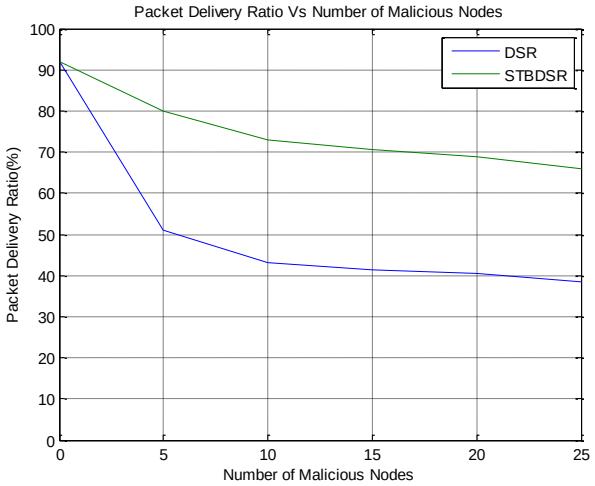



Fig. 6. Comparison of DSR and STBDSR against malicious attacks

## 6 CONCLUSION

In this paper we have described a secure trust model based on evaluation of experience, knowledge and recommendation. We propose a flexible trust model based on the concept of human trust, which provides nodes with a mechanism to evaluate the trust level of its neighbors. The basic idea consists of using previous experiences, knowledge and recommendations of other neighbors to appraise the trust level of other nodes. We introduce the concept of relationship maturity, which allows nodes to attribute more relevance to the recommendations issued by nodes that know the evaluated neighbor for a long time. We analyze through simulations the performance of the proposed model in a mobile multihop network. The simulation results have showed that in the presence of malicious nodes in ad hoc network, the performance of DSR integrated with proposed trust model evaluation mechanism is better than pure DSR in terms of packet delivery ratio.

## REFERENCES

[1]. IETF MANET work group. http://www.ietf.org/dyn/wg/charter/manetcharter.html

[2]. D. F. Macedo, A. L. Santos, J. M. S. Nogueira, and G. Pujolle, "A distributed information repository for autonomic context-aware manets," *IEEE Trans. Netw. Service Management*, vol. 6, no. 1, pp. 45-55, Mar. 2009.

[3]. N. C. Fernandes, M. D. D. Moreira, and O. C. M. B. Duarte, "An efficient filter-based addressing protocol for

autoconfiguration of mobile ad hoc networks," in *IEEE INFOCOM'09*, Apr. 2009.

[4]. B. Ishibashi and R. Boutaba, "Topology and mobility considerations in mobile ad hoc networks," *Ad Hoc Netw. J.*, vol. 3, no. 6, pp. 762-776, Nov. 2005.

[5]. P Narayan, V R. Syrotiuk,"Evaluation of the AODV and DSR Routing Protocols Using the MERIT Tool", InIn proceeding or ADHOC-NOW 2004, pp25-36

[6]. K.Selvavinayaki, K.K.Shyam Shankar, Dr.E.Karthikeyan "Security Enhanced DSR Protocol to Prevent Black Hole Attacks in MANETs" *International Journal of Computer Applications (0975 – 8887) Volume 7– No.11, October 2010*

[7]. Li, Xin; Jia, Zhiping; Wang, Haiyang;"Trust-based On-demand Multipath Routing in Mobile Ad Hoc Networks" IET Information Security

[8]. Sun, Y., Yu, W. ,Han, Z.,and Liu, K.J.R.:'Information Theoretic Framework of Trust Modeling and Evaluation for AdHoc Networks', IEEE Journal on Selected Areas in Communications, 2006, 24, (2),pp. 305-317

[9]. G. Theodorakopoulos and J. S. Baras, "On trust models and trust evaluation metrics for ad hoc networks," *IEEE J. Sel. Areas Commun.*, vol. 24, no. 2, pp. 318-328, Feb. 2006.

[10]. M. E. G. Moe, B. E. Helvik, and S. J. Knapskog, "TSR: Trust-based Secure MANET Routing using HMMs," *Proc. 4th ACM Symposium on QoS and Security for Wireless and Mobile Networks*, Vancouver, British Columbia, Canada, 27-28 Oct. 2008, pp. 83-90.

[11]. Y. L. Sun, W. Yu, Z. Han, and K. J. R. Liu, "Information Theoretic Framework of Trust Modeling and Evaluation for Ad Hoc Networks," *IEEE J. Sel. Areas Commun.*, vol. 24, no. 2, Feb. 2006, pp. 305-317.

[12]. V. Balakrishnnan, V. Varadharajan, U. K. Tupakula, and P. Lucs, "Trust and Recommendations in Mobile Ad Hoc Networks," *Int'l Conf. on Networking and Services*, Athens, Greece, 19-25 June 2007, pp. 64-69.

[13]. S. Marti, T. Giuli, K. Lai, and M. Baker, "Mitigating Routing Misbehavior in Mobile Ad Hoc Networks," *Proc. 6th Annual ACM/IEEE Mobile Computing and Networking*, Boston, MA, Aug. 2000, pp.255- 265.

[14]. J. Sen, P. Chowdhury, and I. Sengupta, "A Distributed Trust Mechanism for Mobile Ad Hoc Networks," *Int'l Symposium on Ad Hoc and Ubiquitous Computing*, 20-23 Dec. 2006. Surathkal, India, pp. 62-67.

[15]. M. Virendra, M. Jadliwala, M. Chandrasekaran, and S. Upadhyaya, "Quantifying trust in mobile ad-hoc networks," in *Proc. IEEE International Conf. Integration Knowledge Intensive Multi-Agent Syst.*, Waltham, USA, Apr. 2005.

[16]. Wei Gong1 Zhiyang You, Danning Chen, Xibin Zhao, Ming Gu, Kwok-Yan Lam, "Trust Based Malicious Nodes Detection in MANET" IEEE 2006.