A Survey on Detection and Prevention of Black Hole Attack in AODV- based MANETs

Yash Pal Singh[1], Dr. P.K Singh[2], Jay Prakash[3]

Computer Science & Engg. Department, Madan Mohan Malaviya Engineering College Gorakhpur (U.P), India

yashcs0117@gmail.com,topksingh@gmail.com,jpr_1998@yahoo.co.in

ABSTRACT: In mobile ad hoc networks (MANETs), nodes usually cooperate and forward each other's packets in order to enable out of range communication. However, in hostile environments, some nodes may deny to do so, either for saving their own resources or for intentionally disrupting regular communications. This type of misbehavior is generally referred to as *packet dropping* attack or *black hole* attack, which is considered as one of the most destructive attacks that leads to the network collapse.
This paper presents a survey of proposed methods of detecting black hole attack against ad -hoc on-demand distance vector routing protocol in mobile ad hoc networks. In a black hole attack, a malicious node answers each route request with a fake reply claiming to have the shortest and freshest route to the destination. However, when the data packets arrive, the malicious node discards them. Several detection methods are described in this paper, and their strengths and weaknesses discussed.

KEYWORDS: **Ad hoc on-demand distance vector routing protocol, mobile ad hoc networks, black hole attack, Packet dropping attack.**

I. INTRODUCTION

Mobile-ad hoc networks (MANETs) are usually formed by a group of mobile nodes, interconnected via wireless links, which agree to cooperate and forward each other's packets. One of the basic assumptions for the design of routing protocols in MANETs is that every node is honest and cooperative. That means, if a node claims it can reach another node by a certain path or distance, the claim is trusted/true; similarly, if a node reports a link break, the link will no longer be used. While this assumption can fundamentally facilitate the design and implementation of routing protocols, it meanwhile introduces a vulnerability to several types of denial of service (DoS) attacks [2], particularly packet dropping attack. To launch such attack, a malicious node can stealthily drop some or all data or routing packets passing through it. Due to the lack of physical protection and reliable medium access mechanism, packet dropping attack represents a serious threat to the routing function in MANETs. A foe can easily join the network and compromise a legitimate node then subsequently start dropping packets that are expected to be relayed in order to disrupt the regular communications. Consequently, all the routes passing through this node fail to establish a correct routing path between the source and destination nodes. Black hole or *sequence number* attack is one of the most common attacks made against the reactive routing protocol in MANETs. The black hole attack involves malicious nodes fabricating the sequence number, hence pretending to have the shortest and freshest route to the destination. Numerous studies have attempted to devise effective detection methods for this attack. The aim of this paper is to investigate seven black hole detection methods within the scope of ad hoc on demand distance vector (AODV) routing protocol.
The paper is organized as follows. Section 2 provides an overview of the route discovery process of AODV protocol and a description of the characteristics of a black hole attack. Section 3 describes several different attack detection methods.
Section 4 presents comparative analysis of the reviewed methods. We conclude with plan for future work in Section 5.

II. BLACK HOLE ATTACK IN AODV

A. Overview of Route Discovery Process In AODV

In a reactive routing protocol, control packets, namely *Route Request* messages, are broadcast by the source node in order to find the optimal route to the destination node. The destination sequence number is an important attribute in *Route Request* that determines the freshness of a particular route.
Upon receiving the *Route Request* packet, a node either:

i) Replies to the source node with a *Route Reply* packet, if it is the destination node or an intermediate node with 'fresh enough' route information to the destination, or
ii) Forwards the *Route Request* packet to its neighbors if it is neither of the above-mentioned nodes.
An intermediate node is deemed to have a fresh enough route to the destination if the destination sequence number in its routing table entry is greater than or equal to the destination sequence number of the *Route Request*. Once the source node receives the *Route Reply*, it establishes a route to the destination. The *Route Reply* message normally has the incremental value of the *Route Request*'s destination sequence number, normally by one [1]. Fig.1 represents a route discovery technique as described above.

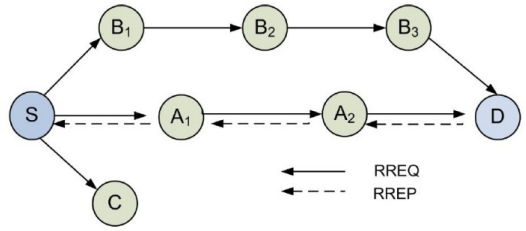

Fig.1. Route Discovery in AODV

B. Black Hole Attack

In on-demand routing protocols, dropping control packets might be the greatly benefit for both selfish and malicious nodes. Specifically, once dropping the RREQ packets, a selfish node prevents the established routes from passing through it and consequently it saves its energy for transmitting its own packets. Likewise, a malicious node can drop the RERR packets in order to prolong the duration of use of the broken routes. As a result, the network throughput collapses sharply since no packet reaches its destination.
A prerequisite for a node to launch a black hole attack is to be involved at least in one routing path. To this end, the malicious node applies the strategies illustrated below.
As shown in Fig.2, C is a malicious node whereas S and D are the source and destination nodes, respectively. First, the node S broadcasts RREQ packet to its one hop neighbors. Then, upon receiving this packet each neighbor node is supposed to rebroadcast it if a route cache towards the destination is unavailable.

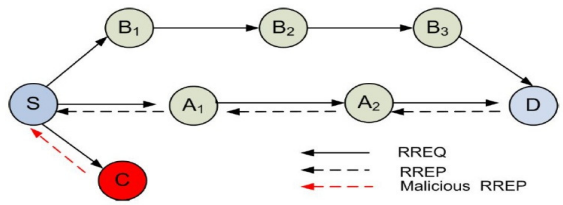

Fig.2. Black Hole Attack in AODV

However, the node C disobeys this rule and claims that it has the shortest path to the destination and sends a RREP packet back to node S. Consequently, if the RREP packet sent by node D or any honest intermediate node, which has a fresh route to D, reaches the node S before the C's RREP then everything works well. Otherwise, the source node S deems that the route passing through the node C is the shortest path, and thus it starts transmitting data packets towards C which in its turn drops them. Another strategy to launch the attack can be described as follows: an intermediate node C spoofs the IP address of the destination D, inciting the source node S to establish the path towards C, instead of D. To illustrate that let us consider the network topology depicted in Fig. 2, when the attacker node C receives a RREQ packet it transmits a RREP packet to reply back to S claiming that it is the intended destination. Moreover, it increases the Destination Sequence Number (Dst-Seq-Num) received in RREQ packet by a value larger than one as shown in Table I, where the node C sets Dst-Seq-Num to 55 rather than 41 to guarantee that the source node S chooses it as the actual destination. The consequences of this attack strategy are similar to the previous one.

TABLE I
THE VALUES OF THE DIFFERENT FIELDS OF RREQ AND RREP PACKETS SENT OR FORWARDED BY BOTH LEGITIMATE AND MALICIOUS NODES: (I) THE NODES *A*1 AND *A*2 FORWARD CORRECTLY THE RREQ AND RREP PACKETS (II) THE NODE *C* SPOOFS THE DESTINATION NODE'S ADDRESS (D) AND AUGMENTS ILLEGITIMATELY THE DST-SEQ-NUM

	RREQ			**RREP**			
Sender	S	A1	A2	D	A2	A1	C
IP-src	S	A1	A2	D	A2	A1	D
Dst-adr	D			S			S
	40			41			55

III. BLACK HOLE DETECTION METHODS

A. METHOD 1: Neighborhood-based Approach

Sun, Guan, Chen and Pooch [3] developed a neighborhood-based approach to detect as well as respond to the black hole attack. The core of their approach is outlined as follows:

1) *Concept*: Once the normal path discovery process is finished, the source node sends a special control

packet to request the destination to send its current neighbor set.

2) *Neighbor set*: The *neighbor set* of a node is defined as all of the nodes that are within the node's radio transmission range. They claim this metric provides a good "identity" of a node, that is if the two neighbor sets received at the same time are different enough, it can be concluded that they are generated by two different nodes. They verified their claim through the following two experiments:
i) They measured the neighbor set difference of one node at different time instants *t* and *t*+1 seconds under different moving speeds and network sizes. The result shows that there is not much change of a node's neighbor set during a route discovery process.
ii) They examined the neighbor set difference of two different nodes at the same time, that is (({A's neighbor set} U {B's neighbor set}) – ({A's neighbor set} ∩ {B's neighbor set})). The result shows that the probability that node A's neighbor set is the same as that of node B is very small.

3) *Detection*: After source node receives the neighbor set information, it analyses them by measuring the neighbor set difference. If the difference is larger than the predefined threshold values, the source node knows that current network has black hole attacks and responds to it accordingly.

4) *Response*: They proposed a *routing recovery* protocol, with the following two-step approach: i) when a black hole attack is identified, the source node uses a cryptography-based method to authenticate the destination, and ii) once verified, the source node sends a control packet to destination node to form a correct path by modifying the routing entries of the intermediate nodes between them.

B. *METHOD 2: Dynamic Training Approach*

Kurosawa, Nakayama, Kato, Jamalipour and Nemoto [4] also adopted an anomaly-based detection technique but incorporated dynamic training technique. In this approach, the *normal state* views are updated periodically to adapt to the frequent network changes and 'clustering-based' technique is adopted to identify nodes that deviate from the normal state. They have adopted the following 5-step process:

1) *Feature Selection*: To express state of the network at
each node, multidimensional feature vector is defined. Each dimension is counted up on every time slot. In order to detect this attack, the destination sequence number is taken into account. In normal state, each node's sequence number changes depending on its traffic conditions. When the number of connections increases the destination sequence number tends to rise, when there are few connections it tends to be increased monotonically. However, when the attack took place, regardless of the environment the sequence number is increased largely. Also, usually the number of sent out RREQ and the number of received RREP is almost the same. From these reasons we use the following features to express the state of the network.
– Number of sent out RREQ messages
– Number of received RREP messages
–The average of difference of Dst Seq in each time slot between the sequence number of RREP message and the one held in the list.
Here, the average of the difference between the Dst Seq in RREQ message and the one held in the list are calculated as follows. When sending or forwarding a RREQ message, each node records the destination IP address and the Dst Seq in its list. When a RREP message is received, the node looks over the list to see if there is a same destination IP address. If it does exist, the difference of Dst Seq is calculated, and this operation is executed for every received RREP message. The average of this difference is finally calculated for each time slot as the feature.
Hence the network state in time slots *i*, is expressed by three-dimensional vector $x_i = (x_{i1}, x_{i2}, x_{i3})$.

2) *Calculate mean*: The mean vector values of these features are calculated, as shown in (1) where *D* represents training data set for *N* time slot.

$$X^{-D} = \frac{1}{N} \sum_{i=1}^{N} x^i \qquad (1)$$

Hence the initial training data refer to the data collected in the first interval of the network, i.e. ΔT_0.

3) *Calculate threshold*: For each time slot, they calculate the distance of each input data sample *x* to the mean vector as shown in (2).

$$d(x) = \| x - x^{-D} \|^2 \qquad (2)$$

From the learning data set, the distance with the maximum value is extracted as threshold *Th*.

$$T_h = d(X_I), \text{ Where } I = \arg_i \max_{X_i \in D} d(X_i) \qquad (3)$$

4) *Anomaly detection*: When the distance for any input data sample is larger than the *Th*, it is considered deviates from the normal traffic and hence, judged as an attack.

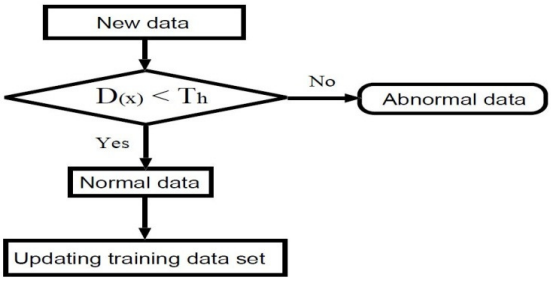

Fig.3. Learning flow chart of proposed method

5) *Dynamic training*: By using data collected in initial time ΔT_0, the calculated mean vector will be used to detect the next period time interval, i.e. ΔT. If the ΔT is judged as normal, the corresponding data set will be used as learning data sets; else, it is treated as data with attack and consequently discarded. This learning process is repeated for every interval ΔT.

C. METHOD 3: Further Request and Reply Approach

H. Deng et.al [5] proposed a solution to cope with the black hole attack in AODV. First, they suggest disabling the ability of an intermediate node to send a RREP and allow only the final destination to do that. This technique avoids the black hole problem but increases the route establishment delay, especially in the case of large networks. Furthermore, since no authentication is used in RREP message a smart attacker can forge a RREP message on behalf of the legitimate destination (by spoofing its IP address). As such, this solution is inappropriate for coping with this attack.

To overcome these shortcomings, they have proposed another solution which requires that the intermediate node adds its next hop's information to the RREP packet before sending it. On receiving this packet, the source node sends a special packet (Further Req) to the next hop of the intermediate node in order to verify that it has a route to the destination and also it is a neighbor of the intermediate node. This special packet contains a field dubbed check result which might be filled by the next hop node. When the source node receives the reply (Further Rep) to this packet it extracts the check result information and decide accordingly whether this route is safe or not. If so, it sends out the data packets, otherwise it initiates a new route discovery or waits for subsequent RREPs. While this solution can avoid the black hole attack launched by a single node, it is unable to detect a collusive attack conducted by both of intermediate and next hop nodes. Moreover, its main disadvantage is the induced overhead if the check process is repeated for each intermediate node replying to the RREQ.

D. METHOD 4: Passive Feedback based Scheme

Watchdog [6] is the first work that has dealt with the problem of nodes which agree to forward packets but never do so. It is designed to secure the DSR protocol and is based on the passive feedback technique, described as follows:

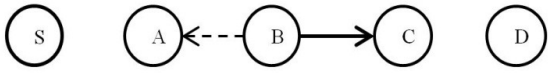

Fig.4. When B forwards a packet from S towards D through C; A can overhear B's transmission and can verify that B has attempted to pass the packet to C. The solid line represents the intended direction of the packet sent by B to C, while dashed line indicates that A is within transmission range of B and can overhear the packet transfer.

1) First, the watchdog node A transmits the packet (p)
originated from S to its next hop B, as shown in the Fig.4.
2) Then it overhears the medium, using the promiscuous
mode [2] to ensure that B has correctly forwarded the packet (p) towards C. If a misbehaving node is identified in the path towards the destination node, then a response mechanism dubbed Path-rater is launched. The goal of path-rater is to establish a new route that avoids the misbehaving nodes.

This scheme suffers from several weaknesses, as stated in [6]. Since a packet collision might occur and prevent the packet to reach the intended receiver, a forwarder node should not immediately be accused of misbehaving, but rather observed for a longer period to make an accurate decision. So, the detection of malicious nodes can take a long time. Moreover, power control transmission and collusion between groups of nodes can trick the watchdog node. Finally, a malicious node can falsely accuse a legitimate node as misbehaving in order to exclude it from the network.

The problem due to collision is described below:

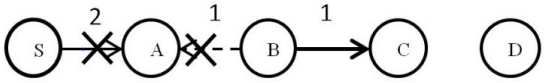

Fig.5. Node A does not hear B forward packet 1 to C, because B's transmission collides at A with packet 2 from the source S

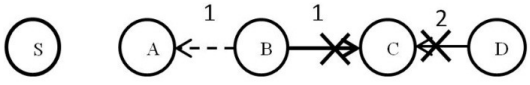

Fig.6. Node A believes that B has forwarded packet 1 on

to C, though C never received the packet due to the collision with packet 2.

Many techniques have been proposed to enhance the robustness of Watchdog. Among them, the work presented in [7] which proposes to choose more than one Watchdog node to avoid the devastating impact of false reports sent by the malicious nodes. To this end, the nodes are classified to ordinary, trusted and Watchdog nodes in terms of their trustworthiness. The trusted nodes are assumed to be the first nodes that initially form the network. The Watchdog nodes are selected periodically from the trusted nodes exclusively. On
receiving the first reply for the route discovery process that has launched, the source node sends out in the secure Watchdog channel a special message to inform the Watchdog nodes about the ongoing transmission. Then, these nodes start monitoring the intermediate nodes connecting the source and destination nodes in order to report any misbehavior. This scheme can indeed detect and isolate the malicious nodes acting alone or in groups, however the induced overhead due to the new control messages is important.
In order to cope with the aforementioned problem of false reports, Ex-Watchdog is proposed in [8]. In this scheme, each node maintains a table containing information about all the paths it is involved in. Each entry of this table stores the instead of malicious nodes. The authors propose the 2ACK scheme to detect malicious links and to mitigate their effects. This scheme is based on 2ACK packet that is assigned a fixed route of two hops in the opposite direction of the received data traffic's route. In this scheme, each packet's sender maintains the following parameters; (i) list of identifiers of data packets that have been sent out but have not been acknowledged yet, (ii) a counter of the forwarded data packets, (iii) and a counter of the missed packets. According to the value of the acknowledgement ratio (Rack), only a fraction of data packets will be acknowledged in order to reduce the incurred overhead. This technique overcomes some weaknesses of the Watchdog/path rater such as: ambiguous collisions, receiver collision and power control transmission.
Both of the previous works remain vulnerable to the attacks launched by group of nodes. To counter these attacks, [9] provides a framework to mitigate the damage caused by the colluding black hole attack in AODV. The proposed technique has a moderate overhead induced by the ACK sent back by the destination during selected intervals of data transfer period. Throughout the data packets transmission, a flow of special packets is transmitted at random intervals along with the data. The reception of these special packets invokes the destination to send out an ACK through multiple paths. The ACK packets take multiple routes to reduce the probability that all ACKs being dropped by the malicious nodes, and also to account for possible loss due to broken routes or congestion in certain nodes. If the source node does not receive any ACK packet, then it becomes aware of the presence of attackers in the forwarding path. As a reaction, it broadcasts a list of suspected malicious nodes to isolate them from the network.

E. METHOD 5: Selfish behavior control mechanism

A selfish node does not want to waste its resources for the benefit of other nodes. Hence, it refuses to forward other's packets but it still uses their services to communicate. To cope up with such behavior, one possible solution is to deprive the selfish node from the services provided by the rest of the network. Therefore, it will be obliged to cooperate. Otherwise it will be isolated from the network and never get its packets forwarded. This class of solutions is also referred to as *Incentive based schemes*.
One of the most reputable works in this category is the model introduced in [10]. This work proposes the use of a virtual currency, dubbed nuglets, as a payment currency in order to motivate each node to forward other's packets. Using nuglets, the authors have proposed two payment models: the Packet Purse Model (PPM) and the Packet Trade Model (PTM). In the former model, the packet sender loads some nuglets in the packet before sending it. The forwarder of this packet earns some nuglets as a payment for the service. If the quantity of nuglets in the packet reaches zero, then it is dropped. In the latter model, as opposed to the former one the packet's final destination rewards the intermediate nodes using its own nuglets. This model can be described as follows:
Each intermediate node earns some nuglets by buying a packet from its previous node for some nuglets and then selling it to the next node for more of nuglets, and the total cost will be paid by the destination. The main drawback of this technique is how to ensure that some nodes do not sell the same packet to more than one neighbor to earn extra money? and how to ensure that each receiver indeed has enough money to pay for the service? To implement both of these models, each node is equipped with a tamper resistant security module that maintains the nuglets counter in order to prevent the nodes from illegitimate increase of their own nuglets.

F. METHOD 6: Reputation based schemes

In this scheme each node must form an opinion regarding the other nodes based on their observed past behaviors. Then the nodes with low reputation are punished or avoided while establishing routes. The major drawback of this category is the excessive traffic exchange needed for sharing the reputation information between the nodes. Moreover, a serious vulnerability of reputation based schemes is the fact that any compromised node can send forged reputation information in order to decrease the trust level of some nodes.

CONFIDANT [11] detects misleading nodes by means of observation and more aggressively informs other nodes of this misbehavior through reports sent around the network. Each node in the network hosts a monitor for observations, reputation records for first-hand and trusted second-hand reports, trust records to control the trust assigned to received warnings, and a path manager used by nodes to adapt their behavior according to reputation information. In more recent work [12] [13], these researchers find that reputation schemes can be beneficial for fast misbehavior detection, but only when one can deal with false accusations, for which they propose a solution using Bayesian statistics. Our goal is to avoid the machinery for managing these reports and their associated trust issues entirely.

CONFIDANT is suitable for small networks with low mobility; however it might be less efficient for large networks since each node needs to maintain a huge table for reputation purposes. Likewise, the high mobility of nodes increases significantly the communication overhead. Additionally, this protocol inherits all the problems of passive-feedback based schemes since it uses this mechanism for the monitoring function.

G. *METHOD 7: Mechanism for controlling cooperative back hole attack*

Jaydip Sen et.al [14] proposed a mechanism for defending against a cooperative black hole attack. The mechanism modifies the AODV protocol by introducing two concepts-

(i)Data routing information (DRI) table and (ii) cross checking. In the proposed scheme, two bits of additional information are sent by the nodes that respond to the Dashed Arrow: RREQ/RREP Packets RREQ message of a source node during route discovery process. Each node maintains an additional data routing information (DRI) table. In the DRI table, the bit 1 stands for 'true' and the bit 0 stands for 'false'. The first bit 'From' stands for the information on routing data packet *from* the node (in the *Node* field), while the second bit 'Through' stands for information on routing data packet through the node (in the *Node* field). The cross checking algorithm proposed in this paper able to deal with the Cooperative black hole attack in AODV based MANETs.

IV. CHARACTERISTIC TABLE OF REVIEWED METHODS

TABLE II

A COMPARATIVE STUDY OF VARIOUS METHOD REVIEWD

	Characteristics					
	Defense Against Collusive attack	Computational Overhead	Communication Overhead	Punishment	Scalability	Latency
D.Sun [3]	N/A	Low	Medium	No	Yes	High
Watchdog[6]	No	Low	No	No	Yes	No
Nuglets (PPM) [10]	N/A	Low	No	Yes	No	No
Nuglets (PTM) [10]	N/A	Low	No	Yes	No	Low
CONFIDANT[11]	No	Low	Low	Yes	Yes	No
Jaydip Sen[14]	Yes	High	Very High	Yes	Yes	High

V. CONCLUSION AND FUTURE WORK

Ad hoc networks are an increasingly promising area of research with lots of practical application. However MANET are extremely vulnerable to Attack due to their dynamically changing topology, absence of conventional security infrastructure and open medium of communication, which unlike their wired counterparts cannot be secure, Selfish or malicious nodes may do intended packet dropping misbehaving This paper has consolidated various works related to black hole attack detection methods in AODV-based MANETs. A comparative study between them was then conducted to highlight their respective effectiveness and limitations. For future work, we plan to develop a more complex black hole attack scenario. In addition, we will construct a detection algorithm to

handle such a complex scenario with an acceptable level of detection accuracy and low computational overhead.

REFERENCES

[1] C.E.Perkins, E.M.B. Royer and S.R.Das, "Ad Hoc On-Demand Distance Vector (AODV) routing", RFC 3561, July 2003.

[2] X. Wu and D. K. Y. Yau, Mitigating denial-of-service attacks in MANET by incentive-based packet filtering: A game-theoretic approach, *In Proc.* 3*rd International Conference on Security and Privacy in Communications Networks*, Nice, France,September 2007.

[3] B. Sun, Y. Guan, J. Chen and U.W.Pooch, "Detecting black-hole attack in mobile ad hoc networks", *Proc. 5th European Personal Mobile Communications Conference*, Apr 2003, pp.490-495.

[4] S. Kurosawa, H. Nakayama, N. Kato, A. Jamalipour and Y. Nemoto, "Detecting blackhole attack on AODV-based mobile ad hoc networks by Dynamic Learning Method", Intl Journal of Network Security, vol 5, no.3, Nov. 2007, pp. 338-346.

[5] H. Deng, W. Li and D. P. Agrawal, Routing security in wireless ad hoc networks, *IEEE Commun. Mag.*, 40(10): 70-75, October 2002.

[6] S. Marti, T. J. Giuli, K. Lai and M. Baker, Mitigating routing misbehavior in mobile ad hoc networks, *In Proc.* 6*th annual international conference on Mobile computing and networking (MOBICOM '00)*,Boston, Massachusetts, USA, August 2000.

[7] A. Patcha and A. Mishra, Collaborative Security Architecture for Black Hole Attack Prevention in Mobile Ad Hoc Networks, *In Proc.*
Radio and Wireless Conference (RAWCON '03), Boston, Massachusetts, USA August 2003.

[8] N. Nasser and Y. Chen, Enhanced Intrusion Detection System for Discovering Malicious Nodes in Mobile Ad Hoc Networks, *In Proc. Internationl Conference on Communication (ICC 07)*, Glasgow, June 2007.

[9] S. S. Ramaswami and S. Upadhyaya, Smart handling of Colluding black Hole attacks in MANETs and Wireless Sensor Networks using Multipath Routing, *In Proc. Workshop on Information Assurance*, United States Military Academy, West Point, NY, 21-23 June 2006.

[10] L. Buttyan and J. P. Hubaux, Nuglets: A Virtual Currency to Stimulate Cooperation in Self-organized Mobile Ad Hoc Networks, Swiss Federal Institution of Technology, Lausanne, Switzerland, Tech. Rep. DSC/2001/001, January 2001.

[11]. S. Buchegger and Jean-Yves Le Boudec. Performance Analysis of the CONFIDANT Protocol; Cooperation of Nodes - Fairness in Dynamic Ad Hoc NeTworks. In *Proceedings of IEEE/ACM Symposium on Mobile Ad Hoc Networking and Computing (MobiHOC)*, Lausanne, CH, June 2002.

[12] S. Buchegger and Jean-Yves Le Boudec. The Effect of Rumor Spreading in Reputation Systems for Mobile Ad Hoc Networks. In *WiOpt'03: Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks*, March, 2003.

[13] S. Buchegger and Jean-Yves Le Boudec. Coping with False Accusations in Misbehavior Reputation Systems for Mobile Ad Hoc Networks. EPFL Technical Report Number IC/2003/31, 2003.

[14] Jaydip Sen[1], Sripad Koilakonda[2], Arijit Ukil[3], A Mechanism for Detection of Cooperative Black Hole Attack in Mobile Ad Hoc Networks, In 2011 Second International Conference on Intelligent Systems, Modeling and Simulation.