Secure AODV against maliciously packet dropping

2 Author(s)

Soleimani, M.T. ; Dept. of Electr. & Comput. Eng., Qazvin Azad Univ., Qazvin, Iran ; Ghasemi, A.

Due to the open shared medium of wireless communications; wireless Ad hoc networks are more vulnerable toward attacks like black hole, which is a kind of packet dropping attack. It is a dangerous type of DOS attacks which try to harm routing protocols. In black hole, the malicious nodes try to absorb all packets in the networks by advertising themselves as having shortest path to the destination. We present a novel approach to detect this attack based on the neighbor's information. In this scheme, we show that the right place to validate route reply and prevent propagation of forged information in the network is the first node in the reverse path. Analysis and simulation results in ns2 show that using the proposed approach, we can successfully detect the black hole attack with a slightly delay on the network.

Published in:
Networked Computing (INC), 2011 The 7th International Conference on

Date of Conference:
26-28 Sept. 2011

Page(s):
5 - 10

Conference Location :
Gyeongsangbuk-do

E-ISBN :
978-89-88678-43-5

Publisher:
IEEE

Print ISBN:
978-1-4577-1129-9

INSPEC Accession Number:
12317705

Secure AODV against Maliciously Packet Dropping

Mohammad Taqi Soleimani
Department of Electrical and Computer Engineering
Qazvin Azad University, Qazvin, Iran
soleimani@qiau.ac.ir

Abdorasoul Ghasemi
Department of Electrical and Computer Engineering
K. N. Toosi University of Technology, Tehran, Iran
arghasemi@eetd.kntu.ac.ir

Abstract— **Due to the open shared medium of wireless communications; wireless Ad hoc networks are more vulnerable toward attacks like black hole, which is a kind of packet dropping attack. It is a dangerous type of DOS attacks which try to harm routing protocols. In black hole, the malicious nodes try to absorb all packets in the networks by advertising themselves as having shortest path to the destination. We present a novel approach to detect this attack based on the neighbor's information. In this scheme, we show that the right place to validate route reply and prevent propagation of forged information in the network is the first node in the reverse path. Analysis and simulation results in ns2 show that using the proposed approach, we can successfully detect the black hole attack with a slightly delay on the network.**

Keywords- Wireless ad hoc network; AODV; packet dropping; black hole attack

I. INTRODUCTION

The Ad hoc On-Demand Distance Vector (AODV) routing protocol [1] is derived from The Destination-Sequenced Distance Vector (DSDV) routing protocol [2] for wireless Ad hoc networks. It is a reactive routing protocol. That is, when each node has some packets to send, it first checks its routing table to find a valid and active path to the particular destination. If there was not any path, then it initiates path discovery phase by broadcasting a route request (RREQ) packet to its all-immediate neighbors. When an intermediate node receives a RREQ packet, if it is the destination of the packet, it sends back a route reply packet (RREP) to the source node through the reverse path. Otherwise, it looks up in its routing table to find any entry that matches to the destination. In case that an entry is found, it checks the freshness of the route by comparing destination sequence number in its routing table to the same one in the RREQ packet. If the sequence number in the routing table is larger than or equal to the sequence number of the packet, it sends back a RREP. The node rebroadcasts the RREQ to its neighbors, if it is not the destination or has not any information about that. (Fig. 1)

Since there is no security mechanism in AODV, this routing protocol is vulnerable to many threats. In AODV, it is assumed every node is truthful. Once a node claims that it has the shortest path to the destination, other nodes may trust it. A severe attack against the routing protocols in wireless ad hoc networks is a black hole attack [3, 4], which is a type of denial of service attack. Malicious nodes can attract all network traffics by falsely claiming to have a fresh and the shortest path to the destination. When a RREQ packet is received by a malicious node, it sends back a RREP packet with a large sequence number and less hop count, which implies a fresh and shortest path to the destination. Once the source node receives the RREP packet, sends all packets to this adversary node as the next hop. However, the malicious node drops all received packets and forms a denial of service attack against the network.

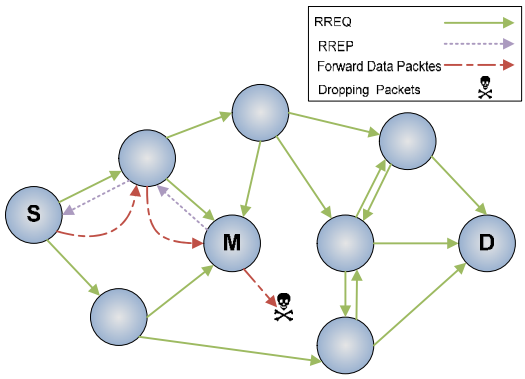

Fig. 1: The adversary node in the network responds any received RREQ by false RREP which it claims having the freshest and shortest path to the destination. When data packets are received, it simply drops them.

To absorb high percentage of the network traffics maliciously, the position of the attacker is an important factor. When an adversary node is positioned near the source node or in a region where normal node density is high, the ratio of packet dropping by malicious node will be increased.

In this paper, we present a new scheme to detect and prevent the black hole attack. Using neighbor node's information could be a powerful tool to find out whether a suspicious node is an adversary or not. In this scheme, once the first node in the reverse path receives a RREP, it checks the validity of the RREP by using the neighbor node's information. Also, we show that, necessary number of nodes which should be deployed in the network is depended on the network field area, maximum transmission range of nodes and number of neighbors that a node should have. In this paper an upper bound for waiting time to receive a correct RREP by the source

node is demonstrated. The main advantage of the proposed scheme among the previous works is that it can detect attacker perfectly with no false alarm. The drawback of the solution is slightly increased delay in the network.

The rest of this paper is organized as follow: Related works on black hole attack detection and prevention is described in section 2. In section 3, the network model, attack model and our solution to detect and prevent the attack will be described. In section 4, we present the simulation results and discuss on them. Section 5 provides our conclusion and future works about the proposed algorithms.

II. Related Works

In [5], authors present a secure version of AODV to countermeasure toward the black hole attack based on the additional control packets, which are called *Further Route Request* and *Further Route Reply*. When an intermediate node sends a RREP to the source node, the source will send a *further route reply* to the next hop of the sender RREP node to verify whether it has really a route to the destination node or not. The node will send back a *Further Route Reply* which contains *checking results*. If the next node has no route to the intermediate node and the destination, the source node broadcasts an alarm packet to the whole network and initiates a new route request. In the case that the next node has no route to the intermediate node but it has a route to the destination, the source node forwards its packets through this node. The main drawback of this scheme is that if the intermediate node is far away from the source, the overall delay of network will be increased.

In [6], authors developed a neighbor-based method to detect black hole attack. Once the routing discovery phase is finished, source node requests the destination to send its neighbor set. After the source node receives the neighbor set information, it calculates the difference between them. If difference is larger than the predefined threshold, it finds that the network is exposed to black hole attack. Then, the source node authenticates the destination node using a cryptography based method and sends a control packet to correct the path.

In [7], Hollic et al. present an analytical model of the AODV route acquisition process. The proposed model can predict the probability density function (PDF) of the estimated route length. The model is extended in [8] to classify node misbehaviors. In this scheme, all destination nodes which are farther than half of the maximum hop count away from the source node will be determined as malicious node probably. The authors claimed that for distance greater than that value, number of correct routes to the destination will decrease and malicious nodes will answer the RREQ packet with a higher probability. In the other words if destination nodes are located far away from the source nodes, the false positive will be raised.

Authors of [9] proposed two solutions to detect the black hole attack. In the first solution, a path will be selected among all received routes, in terms of shared hops. From the shared hops the source node can recognize the safe route to the destination. The main drawback of this approach is to force more delay on the network. In the second solution, each node stores the *last-packet-sequence-numbers* for the last packet sent to each node and the *last-packet-sequence-numbers* for the last packet received from each node. The received RREP contains *last-packet-sequence-numbers* received from the source node. According to the sequence number, the source node can detect the malicious RREP.

In [10], authors proposed a solution to identify black hole attack using anomaly detection. They used three features: the number of RREQ packets, the number of received RREP packets and the average difference between the destination sequence number in the received RREQ packet and a list in the node in each interval. The mean vector $\overline{x}^D$ is calculated as $d(x) = \left\|x - \overline{x}^D\right\|$. Where D represents the training data set for N time slots and x is . When the $\mathrm{d}(x)$, is larger than the threshold T_h, then it will be considered as an attacker. This threshold is taken from $T_h = d(x_I), where\ I = arg_i\ max_{x_i \in D}^{d(x_i)}$. The initial mean vector is calculated in initial time $\Delta T0$, and it will be used to detect the next time interval. If ΔT is judged as normal, the corresponding data set will be used as learning data set. Otherwise, it will be used as data including attack and it will be discarded. This process will be repeated after each time interval ΔT.

Authors of [11] proposed an authentication mechanism based on the hash function, message authentication code (MAC) and pseudo random function (PRF) for detecting black hole attack. In this work, *RREP* packets are signed and encrypted by a sharing secret key. Each node obtains its key by selecting a random number and recursively applying pseudo random function. Also, nodes have to generate a timestamp in the *RREP* packets for validating the packet in the destination node. Each node receives a *RREP* packet; it firstly decrypts and authenticates the packet by its key. Then, it validates timestamp to ensure that it is in a reasonable time delay range. This scheme needs a time synchronization mechanism.

In [12], authors proposed a game theoretic approach for identifying the black hole attack. They used two-player non-cooperative non-zero sum game between MANET and adversary nodes. Also, they introduced a cost function for both players. Each player tries to increase his utility value. They found the Nash equilibrium of the game.

III. BLACKHOLE ATTACK DETECTION ALGORITHM

In this section, we first describe our network and attack model. Then, we present our proposed detection algorithm.

A. Network model

Two different types of nodes are deployed in the network: normal nodes and malicious nodes. All nodes are distributed randomly and uniformly in the square area A. Number of normal nodes in the network is $|N| = n$. It is assumed that all links between nodes are bidirectional and all nodes in the network have the same transmission range R. It is also assumed that each node has no knowledge of entire network topology and it can only discover its one hop neighbors which are located in its transmission range. The nodes in the network have no mobility and each node is identified by a unique identifier throughout the network. We use N_{e_i} to refer as all

neighbors of node i in its transmission range. We also use $d_{max} = 2\sqrt{A}$ to refer as the network diameter. Consequently, maximum hop count in the network is given by $h_{max} = \frac{d_{max}}{R} = \frac{2\sqrt{A}}{R}$. Throughout this paper, we refer to node i as N_i.

B. Attack model

Suppose that M is the set of malicious nodes in the network. Then, the number of malicious nodes in the network is given by $|M| = m$. It is assumed that all malicious nodes are located in the center of the network area. This is a pessimistic assumption. The reason is that when the malicious nodes are located in the center of the network area, they can have more influence. Also, some malicious nodes could be located outside of this region. We called this region as the attacking region. The radius of the this region is denoted as R_m. Therefore, increasing the R_m, can lead to more packet absorption in the network. Also, we assume that each malicious node that receives the RREQ packet responses immediately by sending a RREP packet along with the largest sequence number to the source node, and claims that the destination is one hop far away from itself. It is supposed that the involved region for finding destination node D is a circle area with radius $d + R$, which is centered by source node S. Where d is distance between S and D. As a result, the area of this region is $\bar{A} = \pi(d + R)^2$. If the region $\bar{A}$ has intersection with the attacking area located in center of the square with coordination $(\frac{\sqrt{A}}{2}, \frac{\sqrt{A}}{2})$ and radius R_m, perhaps at least one path among k received route is advertised by an attacker. Thus, if a part or whole area of $\bar{A}$ is in the attacking region, the path will be disrupted by attackers (Fig. 2).

C. Algorithm Description

Once the first node on the reverse path (N_i) receives the RREP, it has to validate the packet to defend against the black hole attack. In order to check the correctness of the advertised RREP, the node broadcasts a *NREQ*[1] packet to all its 2-hop neighbors to find out whether there is a node that has the destination node N_d or suspicious node N_m in its neighborhood or not. In response to this query, each node that is a neighbor of both N_d and N_m sends back a *NREP*[2] packet along with its neighbors list. When a node receives a *NREQ* packet, it searches the blacklist to check whether there is N_m in the list or not. If it is found there, the node immediately sends an alarm packet.

On the other hand, when the destination node receives a *NREQ* packet, it sends a *NREP* packet if N_m is a member of its neighbor set. Otherwise it sends an alarm packet, too.

Those nodes which are only a neighbor of N_m or N_d, will suppose that in the next hop, probably there is a node which has both in its neighborhood. If a receiving node is in the neighborhood of the destination, it relies on the destination to check and forwards the *NREQ* instead of sending *NREP* packet.

[1] NREQ: Neighbor Request

[2] NREP: Neighbor Response

After Δt, if N_m advertised a path correctly, at least one *NREP* packet which contains N_d and N_m should be received by N_i. Once the RREP is validated by other nodes, N_i removes the corresponding RREP from the queue and forwards it to the source node N_s. If there is no received *NREP*, it will be supposed that N_d is located far away from N_m, and N_m is an adversary node (we assumed that the only reason for packet dropping in the network is malicious nodes). As a result, N_i considers N_m as a malicious node, and then drops the corresponding RREP from the queue and keeps N_m in the blacklist. Then, it broadcasts an alarm for all two hops neighbors and indicates N_m as malicious. The main reason of controlling the alarm broadcast is to avoid increased network overhead.

The major problem of the proposed scheme is slightly increased delay in the network. Also, in the cases where an attacker is a neighbor of the destination, when N_i sends *NREQ* and asks N_d that whether N_m is its neighbor or not, N_d will response positive. Then N_i trusts the advertised RREP and forwards it to the source node. When the source forwards all packets to the malicious node N_m, instead of delivering packets to the destination, it simply starts to drop them. Therefore, in this scenario, the proposed scheme will be failed.

To defend against this situation, we use the sequence number. When an adversary node sends a RREP it tries to advertise a larger sequence number to forge the source node as having the freshest path to the destination. Once, N_i tries to identity N_m, it puts the advertised sequence number by adversary into the NREQ packet. Receiving NREQ by destination node, it checks whether N_m belongs to its neighbor list or not. If yes, it compares sequence number of its own with the one in the received packet. If the advertised sequence number is greater than its own, it can detect that N_m is an attacker. Then it prepares an alarm packet and sends back to N_i.

We assume that adversary nodes can not identify and forge the NREQ, NREP and alarm packet.

D. An upper bound for the threshold

In this part we discuss about the upper bound for the threshold that a node should wait to receive a correct *RREP* from the network.

Suppose that d is the distance between S and D and v is the speed of the light. The minimum time that source node should wait to receive a route replay from the destination after sending route request is computed as:

$$\left.\begin{array}{l} \Delta t = t_{SD} = \frac{2d}{v} \\ d = hR \end{array}\right\} \Rightarrow t_{SD} \simeq 2hR \quad (1)$$

where t_{SD} is the *RTT* (*Round Trip Time*).

The source node should wait at most t_{SD} to receive RREP from D after sending RREQ. t_{SD} is the upper bound for the source node to wait for RREP. Therefore, the first node in the reverse path should wait $t_{min} = \frac{2d}{v}, h = 2 \Rightarrow t_{min} \simeq 4R$ to receive the NREP packet.

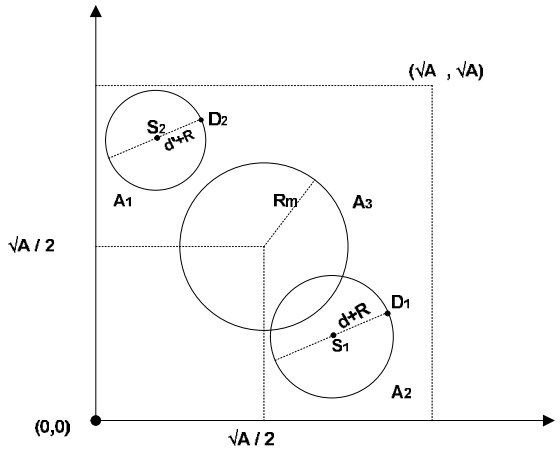

Fig. 2: In the area A_3, density of malicious nodes is more than other network region. Since area A_2 has intersection with A_3, all communication in that region will be controlled by attackers. While, S_2 in the region A_1 is in the safe zone.

The node density in the network is an important parameter, to obtain that value for the threshold.

Supposing that nodes are distributed uniformly, the average of node's neighbors in the network is calculated as: $N_{e_i} = \frac{(n-1)}{A} \pi R^2$ [13].

If the node density in the network is less, the end to end delay will be increased, since nodes may be located with distance R from each other. In the common area of node x and y (Fig. 3), at least Υ ($\Upsilon \geq 1$) nodes should exist.

According to the Fig. 3, we have: $\cos\theta_1 = \frac{r/2}{r} = 0.5$. Therefore, $\theta_1 + \theta_2$ can be computed as $\theta_1 + \theta_2 = \frac{\pi}{3}$. We refer $\bar{\bar{A}}$ as the intersection area of two circles. That is: $\bar{\bar{A}} = 2R^2\left(\frac{\pi}{3} - \frac{\sqrt{3}}{4}\right)$. The minimum number of nodes that should be deployed in this area is: $\frac{\bar{\bar{A}}}{A} \times n \geq \Upsilon$. Therefore, the minimum number of nodes in the area A with transmission range R to ensure that each node has averagely Υ neighbors is given by:

$$n \geq \frac{\Upsilon A}{\left(\frac{\pi}{3} - \frac{\sqrt{3}}{4}\right)R^2} \quad (3)$$

Increasing Υ causes the waiting time to be closed to t_{min}. In figure 4, source node S initiates the route discovery phase. The node (M) claims having the shortest path, the first node in the reverse path X, should validate the received route reply. The value of Υ in the Fig. 4 is zero. Once node X received a RREP, it should request its neighbors except node M, to determine which node has node M and D in its neighborhood. The shortest path between X and D that doesn't include M, is 4 hops. Hence, to validate the RREP by X, it has to wait 8 units of time in terms of hop count. Consequently, the imposed delay of the network will be increased. Therefore, waiting time will be greater than t_{min}. By increasing Υ, the shortest path between X and D, which doesn't involve M, can be 2 hops. And the imposed delay will be 4 units of time in terms of hop count. Hence, as increasing Υ, the imposed delay on the network will be decreased and it will be closer to t_{min}.

IV. SIMULATION AND DISCUSSION

This section reports the simulation results on proposed scheme. We've used NS-2 [14] (Network Simulator) to simulate our network and AODV is used as the routing protocol. The simulation parameters are summarized in table 1. The simulation is done to evaluate the performance of network parameters. Considered metrics are as below:

- Packet Delivery Ratio: implies the packets that are sent from source node and delivered to the destination.
- Malicious Detection Rate: implies number of malicious nodes detection when a faked RREP is sent back in order to form a DOS attack.
- Average End-to-End Delay: The time difference between sending a packet from source node to the destination node. It includes all the delays from route discovery phase, transferring and queuing in the source node and intermediate nodes.
- Average Energy Consumption Ratio: This is the average ratio of energy consumed by each node at the end of simulation.

The result in the Fig. 5 compares the packet delivery ratio with the number of the attackers in the network. When there is no attacker in the network, normal AODV routing protocol has packet delivery ratio of 1. Since, we assumed that the only reason for packet dropping in the network is due to the malicious nodes. We ignore all other reasons. As the number of attackers increases, the delivery ratio decreases. Using the proposed scheme, the delivery ratio is still 1 and the packet dropping is kept zero. The malicious nodes can be detected by the proposed scheme correctly if network is under attack. Also, when there is no adversary node in the network, the proposed scheme has no miss detection. This is the pivot point and major consequence of the scheme.

We compare our solution with the proposed schemes in [5, 6]. To evaluate the schemes in terms of the imposed delay, we compare the end-to-end delay versus the number of flows. As it is illustrated in the Fig. 6, the overall delay of our scheme is closer to the delay of AODV. While, the imposed delay of the works [5, 6] are greatly increased.

The reason is that, once the RREP is received by the source, it starts validating by asking from the next hop of the sender RREP node. Since when the first node in the reverse path is responsible to validate the advertised route, source node has to wait more.

Another advantage of our scheme is to avoid propagating false route in whole network. When the first node in the reverse path determines the advertised route is false, it will drop it to prevent node to learn forged routes. If the received route is faked in solution [5, 6], the source node has to alert all nodes in the network to block the adversary node and correct their routing table by broadcasting an alarm packet. Also, that solution cannot detect attack either in presence of the multiple attackers or malicious node is in adjacent to the destination node. However, in our scheme, it is not necessary to send the alarm packet to whole network. As it is shown in figure 6, by

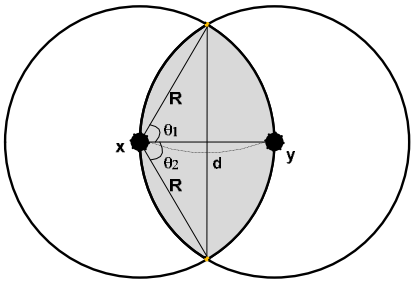

Fig. 3: Shadowed area indicates the intersection of the two circles with radiuses R which located in center of them. If there are enough nodes in this area, then detecting of black hole is done easily.

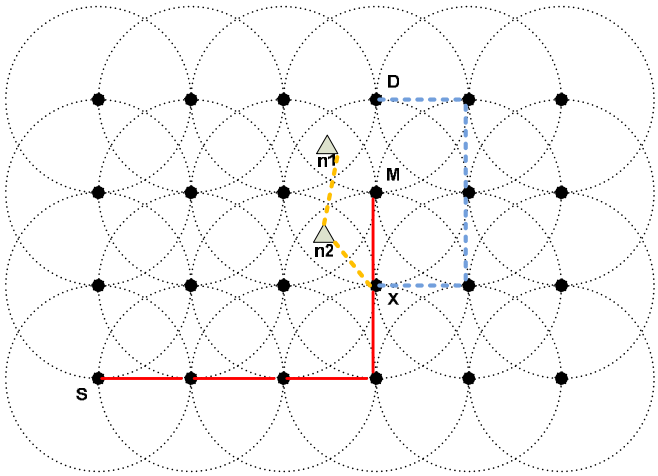

Fig. 4: The value of ϒ is zero. Node X has to wait has to wait 8 units of time in terms of hop count to validate the received RREP from M. The dotted line shows the shortest path between X and D which node M is not belong to the path. By adding two nodes n1 and n2, the waiting time is decreased.

increasing number of adversary nodes, our solution can detect the attackers with a negligible- increase delay by comparing to the scenario where there is one attacker in the network.

Average energy consumption is displayed in the figure 7. As it shows when our scheme is applied and network is under attack, average energy consumption in the network has increased insignificantly. That is because for sending and processing additional packet such NREQ and NREP packets. Since, the broadcasting the alarm packet is controlled; the network overhead is greatly decreased. As a result, energy consumption of nodes in the network will be kept low. Therefore, this scheme can be used in those networks where the energy is a major constraint.

According to the discussed advantages of our scheme, the right place to validate the advertised route by an intermediate node is the first node in the reverse path to prevent spreading false route in the network.

V. Conclusion and Future Works

In this work we studied the AODV routing protocol and black hole attack. We presented a novel approach to prevent the maliciously packet dropping with considering the number of neighbor each node should have. Also, we show that, the right place to validate the RREP which it is sent by an intermediate node should be the first node in the reverse path, to avoid propagating false route information in the network. Future work includes extending this work for MANET. Also, we would like to extend the proposed scheme for detecting the wormhole attack.

Table 1: Simulation Parameters

Parameter	*Value*
Routing Protocol	AODV
Mac	IEEE 802.11
Terrain Area	$1000m \times 1000m$
Transmission Range	$200m$
Number of Nodes	200
Number of Attackers	8
ϒ	5
Traffic Type	CBR
Packet Size	512 Kb
Rate	100 Kb/s
Number of Flows	5
Simulation Time	1000 seconds

References

[1] C. E. Perkins and E. M. Royer. "The Ad Hoc On-Demand Distance Vector Protocol," in Proc. C. E. Perkins (Ed.), Ad Hoc Networking, pp. 173–219. Addison-Wesley, 2000.

[2] C. E. Perkins and P. Bhagwat. "Highly Dynamic Destination-Sequenced Distance-Vector Routing (DSDV) for Mobile Computers," in Proc. SIGCOMM 94: Computer Communications Review, 24(4), pp. 234–244, October 1994.

[3] C. Karlof and D. Wagner, "Secure Routing in Sensor Networks: Attacks and Countermeasures," in Proc. 1st IEEE International Workshop on Sensor Network Protocols and Applications, (SNPA 2003), pp. 113-127, May 2003.

[4] L. Abusalah, A. Khokhar, M. Guizani, "A survey of secure mobile Ad Hoc routing protocols," Communications Surveys & Tutorials, IEEE, vol 10, issue 4, pp. 78-93, 2008.

[5] H. Deng, W. Li; D.P. Agrawal, "Routing security in wireless ad hoc networks," IEEE Communications Magazine, vol 40, Issue 10, pp.70 - 75, Oct 2002

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

[7] M. Hollick, J. Schmitt, C.Seipl and R.Steinmetz, "The ad hoc on demand distance vector protocol: an analytical model of the route acquisition process," in Proc. of Second Intl Conference on Wired/Wireless Internet Communications (WWIC'04), Frankfurt, pp. 201-212, Feb 2004.

[8] M. Hollick, J. Schmitt, C. Seipl and R.Steinmetz, "On the effect of node misbehavior in ad hoc networks," in Proc. Of IEEE Intl Conference on Communications (ICC'04), Paris, pp. 3759-3763, June 2004.

[9] Al-Shurman, M. Yoo, S. Park, "Black hole attack in Mobile Ad Hoc Networks," in Proc. ACM Southeast Regional Conference, pp. 96-97, 2004.
[10] S. Kurosawa, H. Nakayama, N. Kato, A. Jamalipour and Y. Nemoto, "Detecting black hole attack on AODV-based mobile ad hoc networks by Dynamic Learning Method," Intl Journal of Network Security, vol 5, no. 3 , pp. 338-346, Nov. 2007.
[11] J. Luo, M. Fan, and D. Ye, "Black Hole Attack Prevention Based on Authentication Mechanism", in Proc. of 11th IEEE Singapore Intl Conference on Communication Systems (ICCS 2008), Singapore, pp. 173-177, 2008
[12] E. A. Panaousis, C. Politis, "A Game Theoretic Approach for Securing AODV in Emergency Mobile Ad Hoc Networks", in Proc. of 9th Intl Workshop on Wireless Local Networks (WLN 2009), Zürich, Switzerland, pp. 985-992, 2009.
[13] J. C. Hou and N. Li, "Topology Construction and Maintenance in Wireless Sensor Networks", Chapter 10 of Handbook of Sensor Networks: Algorithms and Architectures, John Wiley & Sons, Inc., 2005
[14] Network Simulator 2. http://isi.edu/nsnam/ns/.

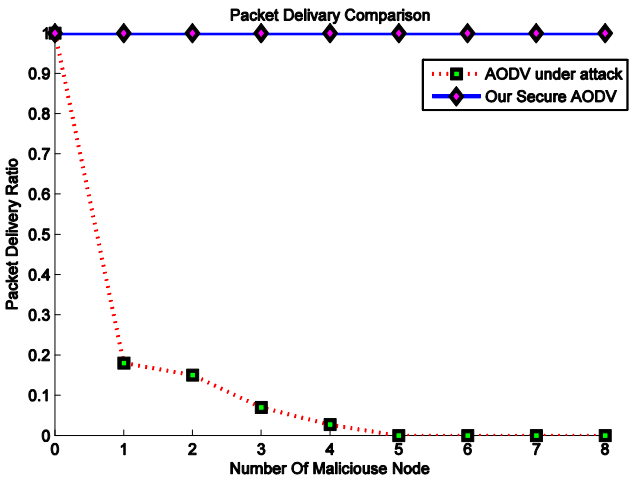

Fig. 5: Packet delivery ratio, when the number of adversary nodes varies.

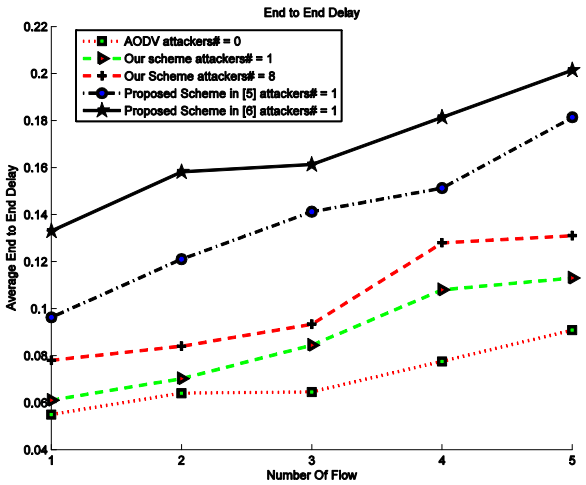

Fig. 6: Average end to end delay, when the number of flow varies.

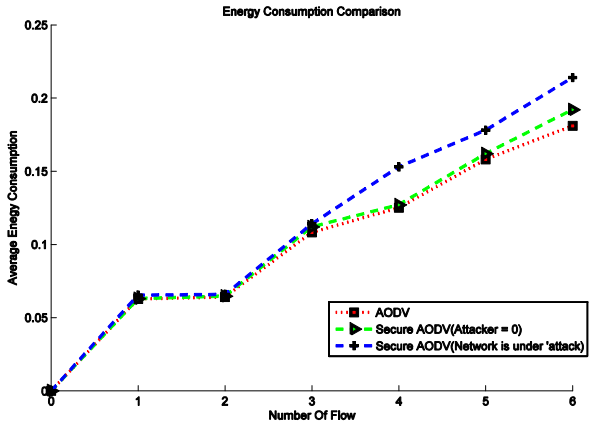

Fig. 7: Average energy consumption, when the number of flow varies.