Fast and Efficient Countermeasure for MAC Layer Misbehavior in MANETs

Soufiene Djahel, Zonghua Zhang, Farid Naït-abdesselam and John Murphy

Abstract

In this paper, we deal with backoff cheating technique in IEEE802.11 based MANETs and propose a novel scheme, dubbed HsF-MAC (**H**as**h** **F**unction based **MAC** protocol), to cope with it. In contrast to the existing solutions, HsF-MAC allows MANET nodes to re-calculate the backoff value used by their 1-hop neighbors and immediately detect the misbehaving ones. Moreover, the colluding behavior of two cheating nodes is also considered along with effective countermeasures. A reconciliation based reaction mechanism is finally designed. The simulation results, under different topologies and network conditions, have validated the effectiveness of HsF-MAC.

Keywords – MAC Layer Misbehavior, IEEE802.11, Greedy behavior, CSMA/CA, MANETs.

I. INTRODUCTION

MANETS potentially suffer from a range of vulnerabilities due to their special characteristics, such as shared wireless medium, limited energy resources, rapidly changing topology and untrustworthiness of the partners. These vulnerabilities can be easily exploited by a misbehaving node to launch a bunch of attacks at different layers, especially at MAC layer where the manipulation of DCF parameters does not need as much effort as the attacks launched at higher layers. A misbehaving node disobeys the protocol rules to gain extra bandwidth au detriment of the neighbor honest nodes. To do so, it may use several cheating techniques, however this paper's focus is on backoff cheating technique in which the misbehaving node sets its backoff to a small value rather than calculating it following the Binary Exponential Backoff (BEB) algorithm.

Since IEEE 802.11 [2] MAC protocol is commonly used by wireless nodes to access the shared medium, any misbehavior at this level alters the proper functioning of the network. In the last decade, many researchers have focused their efforts on investigating the root causes of MAC layer misbehavior, such as the work done in [3]. Based on the findings of these studies, various approaches have been proposed in the literature to design a resistant MAC protocol that is able to either prevent or detect any misuse at MAC level. These approaches can be split into two main categories as follows. The first category of

Soufiene Djahel and John Murphy are with the Department of Computer Science and Informatics, UCD, Ireland (soufiene.djahel, j.murphy)@ucd.ie).
Zonghua Zhang is with Institute Mines-Telecom/TELECOM Lille1, France (zonghua.zhang@telecom-lille1.eu).
Farid Naït-abdesselam is with University of Paris Descartes, France (naf@parisdescartes.fr).
Manuscript received March 2012.

solutions consists usually of a trustworthy central node (e.g. WLAN Access Point or a Wireless Mesh Router) that monitors the behavior of its neighbors during equal-length time intervals and then uses several mechanisms (e.g. simple tests [5], fuzzy logic based tests [10] or statistical methods [6]) to analyze the collected information and detect any misbehavior. In the second category, the backoff algorithm is changed in most solutions in order to make the backoff value predictable or give the receiver node a control over it. For example, the backoff value to be used by the sender node is assigned by its intended receiver through a long negotiation process, like the work done [4], and the interval from which a sender randomly chooses a backoff value is dynamically changed based on the previous chosen value rather than the transmission status as in [9], which leads to low channel utilization.

The major drawbacks of the previous approaches are that they either require a large number of observation samples to detect the cheaters or trigger a non-negligible number of false alarms, which compromises their accuracy. Additionally, most of these approaches do not provide a robust countermeasure to cope with the collusive misbehavior[1]. To mitigate the above weaknesses, we propose a proactive defense scheme, dubbed HsF-MAC (**H**as**h** **F**unction based **MAC** protocol), that takes actions at earlier stage to efficiently deal with MAC layer misbehavior launched by both single and colluding nodes. Moreover, we propose a novel reaction scheme to penalize the detected cheaters.

A detailed description of the proposed detection and reaction schemes is presented in next section. Afterwards, we present and discuss the simulation results in section III. Finally, we conclude this paper in section IV.

II. The proposed scheme

Coping with the greedy behavior at MAC layer in MANETs remains as a grand challenge despite the numerous efforts. To solve this problem, we have designed a novel scheme that aims to achieve the following objectives: remove the assumption of trustworthiness of one part of the communication, detect the nodes which either maliciously choose a small backoff value or refuse to double their CWs (Contention Window) after collision, ensure fast detection of both single and colluding cheaters (i.e. sender and receiver) with less number of false alarms. For the sake of simplicity, we assume that an anti-MAC spoofing mechanism is set up at MAC layer and that the number of cheating nodes within a neighborhood is less than the number of honest nodes.

In HsF-MAC, two new fields dubbed $Attempt$ (Γ) and CRC are added to each RTS frame as shown in Fig. 1. Γ represents the number of times the sender has tried to transmit the RTS frame and its corresponding data packet, whereas CRC is the Cyclic Redundancy Code calculated over all the fields of the data packet to be transmitted. The (Γ) value is initialized to 1 after a successful transmission and incremented by 1 for each unsuccessful transmission of RTS or data packet. Using these two parameters, the backoff value is computed according to the following formula:

[1]A collusive misbehavior is driven by two misbehaving nodes (e.g. a sender node and its corresponding receiver) that support each other to increase their bandwidth at the expense of their neighbors.

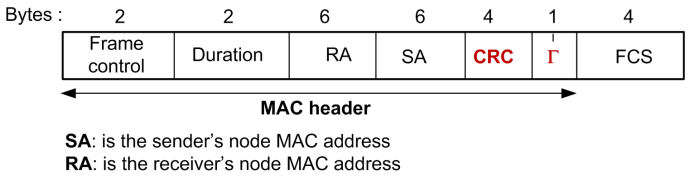

Fig. 1: The new format of the RTS frame

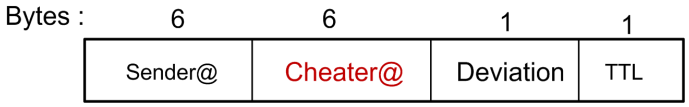

Fig. 2: Warn-single packet format

$$backoff = Hash(fct(CRC, \Gamma)) mod\, 2^{\Gamma-1} CW_{min} \tag{1}$$

where

$$fct(CRC, \Gamma) = (CRC \oplus \Gamma) \tag{2}$$

Here CW_{min} is the minimal contention window which varies according to the technology used at physical layer (e.g. CW_{min} = 31 for DSSS and 15 for OFDM). The CW value is doubled after each collision. It is worth noting also that $Hash$ refers to any one way function like MD5 and SHA1.

By adopting this scheme the receiver can detect any deviation from the sender since it is able to recalculate the backoff value that has been chosen by this sender. This is feasible due to the following features of HsF-MAC.

1) The CRC and Γ values used by the sender to compute its backoff value are known to the receiver.
2) The purpose of using hash function to generate the backoff value is to prevent any node from setting its backoff to a small value since it is generated by one way function (i.e. it is hard to find the inputs that produce a specific small value).
3) Using the CRC field as an input to the fct function may significantly reduce the number of collisions since its value differs in each data packet.
4) We have added Γ as a second input to the fct function in order to deal with retransmissions; as in this case CRC is unchanged, the new backoff value should be different from the previous one because Γ increases with every failed retransmission attempt.

To summarize, HsF-MAC guarantees the randomness property of the regular backoff scheme, thanks to the hash function, and keeps the likelihood of repetitive collisions to minimal due to chosen inputs of the fct function.

In what follows, we describe in details how HsF-MAC detects a single cheater? How this cheater is get penalized? And how HsF-MAC deals with a collusive misbehavior of two cheaters?

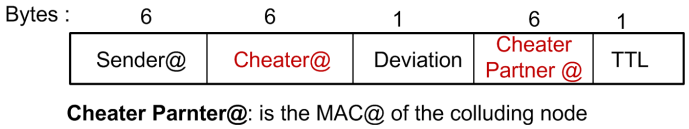

Fig. 3: Warn-collusion packet format

A. How a cheater node is detected?

On receiving an RTS frame, the receiver node extracts the CRC and Γ values from this frame and uses them together to re-calculate the backoff value used by the sender, following the formula in Equation (1). If the number of elapsed idle slots [2] (excluding the slots spent for other neighbor nodes' transmissions) after the last transmission of the sender is less than ((backoff + DIFS)- ϵ), this indicates that the sender node has violated the protocol, thus it deserves an appropriate punishment. We denote that ϵ, called accuracy factor, is used to minimize the number of false alarms. These false alarms are generated as a result of the inaccuracy of all observations taken by the receiver node regarding the backoff used by the sender. This inaccuracy occurs, for example, when the receiver node presumes that the sender has frozen its backoff timer due to a transmission or interference from a third node whereas actually this sender is counting down its backoff.

In case of consecutive collisions incurred by the collision of either CTS or data frames, a cheater node may follow the backoff computation scheme but intentionally keeps the Γ value unchanged and consequently refuses to double its contention window. This misbehavior can be easily detected by HsF-MAC through a comparison of the CRC values corresponding to respective RTS frames. That says, if these values are equal and Γ is unchanged then the receiver node infers that the sender is a cheater. For more sophisticated scenario where the cheater alters also the CRC value to mislead the receiver node, a simple comparison of the CRC value received in the RTS frame with that calculated over the received data frame will reveal the misbehavior of this cheater.

B. Our reaction scheme: warn or punishment?

In contrast to the existing reaction schemes in the literature, our reaction mechanism gives a chance to the cheating node to repent and abide again to the protocol rules, so as to avoid being punished. We have chosen this mechanism because we believe that it is better to incite a cheater node to behave correctly rather than excluding it from the network. This is because in MANETs, every node contributes significantly to ensure the availability of network's services such as connectivity, routing and Internet access. If the cheater node is the only node in the network providing connection to some nodes or it is a gateway to the Internet, then its exclusion from the network may lead to network partition and unavailability of some services like Internet connection. Even though several nodes may provide the same service as the cheater node, its punishment potentially leads to overloading the other nodes if a load balancing mechanism is not in position.

One of the main challenges related to cheater nodes advertisement in MANETs is the prevention of false alarms that severely

[2]We consider only the number of the elapsed idle slots during which no collision is observed.

affects the network performance. These false reports are usually issued from adversary nodes claiming that a given legitimate sender does not comply with the protocol, which leads to its exclusion from the network. In what follows, we describe the functioning of our reaction mechanism and show how it deals with the aforementioned issue.

Once a deviation of a sender node is observed by its corresponding receiver, this latter sends a special message, dubbed Warn-single, which advertises the MAC address of the cheater node as well as the observed deviation from the calculated backoff value. The format of this message is shown on Fig. 2 where the TTL value is set to 4. This value is deemed to be sufficient to allow all the neighbors of the cheater node to receive the $Warn_{single}$ message. Upon reception of this message, each node starts monitoring all the transmissions initiated by the advertised node in order to check whether it is correctly applying the MAC protocol rules or not. Here the advertisement of the sender's identity is considered as a warning message informing this sender that its direct receiver has been aware of its misbehavior and thereby incites it to change its behavior by calculating its backoff value according to the protocol rules. If it does so for its subsequent transmissions, then no punishment will be applied against it, otherwise all its neighbors will punish it.

The one hop neighbors of the cheater node would be aware of its misbehavior as follows. The advertiser of the Warn-single message sets the TTL value to m (m equals 4 in our simulation) which is decreased at each hop. Upon reception of this message, if the receiver node is a neighbor of the cheater node so it discards the message, otherwise it re-broadcasts it (and decreases the TTL value) except if the TTL reaches 0. To penalize a confirmed cheater node, each of its neighbors may refuse to respond to its communication requests (i.e, ignore its RTS frames) or delay the delivery or forwarding of its data packets, which leads to severe performance degradation at the cheater side. Further, those neighbors may also launch a cross layer punishment scheme by refusing to relay any control message sent by the cheater node, so it can never be involved in a routing path.

C. How colluding cheaters are detected?

Sender-receiver collusion is a more sophisticated scenario of MAC layer misbehavior that most of the current countermeasures do not cope with it. This misbehavior provokes a significant reduction of the throughput acquired by the neighbors of the colluding nodes. To defend against this devastating misbehavior, we propose to extend our detection scheme as follows. Whenever a node experiences an unusual decrease of its throughput, it starts supervising all the transmissions in its vicinity to identify which nodes are getting a higher throughput. After identifying those nodes, the monitor node supervises carefully every transmission originated from them and checks whether they comply with the backoff computation rules or not. If any node disobeys our proposed scheme and its intended receiver didn't inform its neighbors about this misbehavior, the monitor node then piggybacks both of their identities in a special message, dubbed Warn-collusion, along with the corresponding estimated deviation, which is the difference between the backoff value calculated according to the Equation. 1 and the observed one.

Notice that the receiver identity is also advertised to flag that it refuses to reveal the identity of a cheating node. To be able

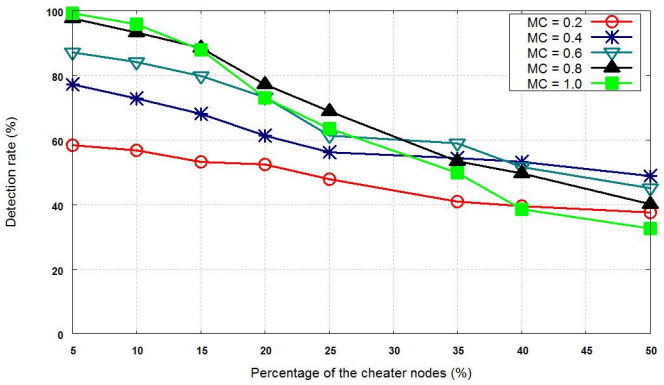

Fig. 4: Detection ratio versus the percentage of the cheater nodes: with random topology

to detect this misbehavior, the monitor node should be either a neighbor of both the sender and receiver or collaborate with another node such that each of them is a neighbor of one of the cheating nodes. A TTL value is also included in the advertised message (see Fig. 3) and its value is set to 8 in order to make all the neighbors of both nodes aware of their misbehavior. These neighbors will then monitor the behavior of the advertised nodes until either their misbehavior is confirmed or they have repented and complied again with the protocol. If their misbehavior is confirmed then the same punishment scheme (i.e. that described in the previous section) is applied.

D. Comparative study

We compare, in Table I, HsF-MAC with the most significant works in the literature based on several criterions. These criterions are chosen to answer the following questions. How fast is a given solution in terms of the number of the required observations to detect a cheater node? Does it provide any reaction mechanism to punish the detected cheaters? Is it resistant to the adaptive cheater which combines several cheating techniques to escape from detection? Is it able to detect two colluding cheaters? Does it ensure that the chosen backoff value is randomly picked? Does it still efficient in MANETs?

Note that due to the pages limitation, we will not provide a detailed description of the existing works, however the reader may refer to our previous work [10] which gives more comprehensive presentation.

III. Simulations and performance evaluation

To evaluate the effectiveness of HsF-MAC, we have conducted a set of simulations using OPNET and taking into account various network sizes and topologies. Wireless nodes are randomly moving, at a speed of 5m/s following random waypoint mobility model, within the network topology and exchanging 500 bytes CBR packets. We use MD5 (128bits) as a hash function to implement HsF-MAC. The setting of the simulation parameters is summarized in the Table II.

Table III shows the Fairness index [1] (Fi) values measured on a random topology with varying network size. Both HsF-MAC and the standard BEB show similar values which are always close to 1. This proves that using a hash function to generate the Backoff value ensures fair share of bandwidth among the contending nodes. The $Norm - thr$ (Normalized throughput)

	Reaction	Detection speed	Adaptive cheater detection	Sender-receiver collusion detection	Ensure the Randomness property	Efficient in MANETs
HsF-MAC	Yes (Distributed)	One transmission	Yes	Yes	Yes	Yes
DOMINO [5]	Centralized	Several monitoring windows	No	N/A	Yes	No (WLAN only)
FLSAC [10]	Centralized	Several monitoring windows	Yes	N/A	Yes	No (WLAN only)
PRB [9]	No	One transmission	Yes	N/A	No (change CW intervals)	Yes
[7]	Centralized reaction	Several monitoring windows	Yes	N/A	Yes	No (WLAN only)
[8]	Yes	Several monitoring windows	No	N/A	Yes	No (WLAN only)
[4]	No	One transmission	Yes	No	Yes	Yes

TABLE I: Comparison of the main features of HsF-MAC with those of the existing schemes

Parameters	Values
No. of nodes	5 . . . 150
Physical layer	Direct sequence
Transmission range	250 m
Carrier sensing range	550 m
Data rate	3 mbps
Traffic type	CBR (500 bytes per packet)
Nodes speed	5m/s
Simulation time	300 seconds
No. of simulation epochs	5

TABLE II: Simulation settings

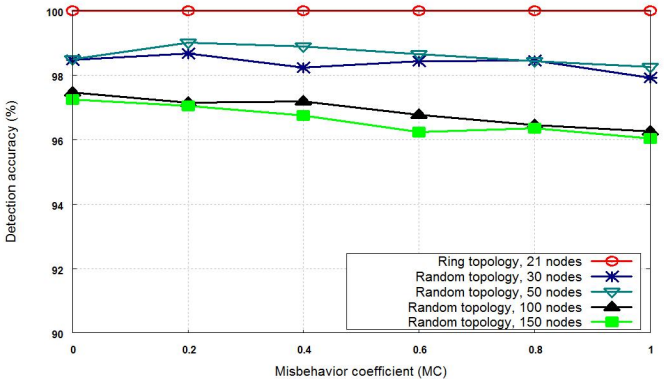

Fig. 5: Impact of the MC, network size and topology on false alarms

No. of senders	5	10	20	30	40	50
F-index of BEB	0.987	0.906	0.968	0.96	0.981	0.968
F-index of HsF-MAC	0.986	0.964	0.951	0.957	0.938	0.964

TABLE III: Impact of the number of senders on the fairness index in a random topology: standard BEB vs. HsF-MAC

of each sender node is presented in Table IV from which we can clearly see that the achieved normalized throughput is much higher in case of small number of senders (dense network of 5 senders) and decreases to the half when the number of senders is doubled. This is due to the fact that in this latter case each node acquires half of the bandwidth gained in the previous scenario. As the wireless nodes are randomly distributed in the network then they constitute separated dense sets of nodes connected among them. Thus, the throughput acquired in each set doesn't affect that of the other sets. So, when the number of senders overtakes 10 nodes the obtained $Norm - thr$ for each node decreases slightly compared to the values obtained in the scenario of 10 senders. Notice that the standard BEB outperforms HsF-MAC in some scenarios however the gap is usually very small.

Table V shows that Fi of HsF-MAC slightly decreases as the offered bandwidth increases and the same impact is observed for the standard BEB. This decrease is justified as follows: when the offered bandwidth is low (1 and 2 mb/s or even 5.5 mb/s), the difference between the earned bandwidth of the nodes is very small and even negligible, however when the offered bandwidth gets higher, such as 11mb/s or larger, the gap between the gained bandwidth of the nodes rises and consequently

No. of senders	5	10	20	30	40	50
Norm-thr in BEB	0.109	0.063	0.063	0.062	0.059	0.055
Norm-thr in HsF-MAC	0.11	0.064	0.057	0.060	0.056	0.056

TABLE IV: Impact of the number of senders on the normalized throughput (% per node) in a random topology: standard BEB vs. HsF-MAC

Offered bandwidth (mb/s)	1	2	5.5	11
F-index of BEB	0.999	0.976	0.961	0.928
F-index of HsF-MAC	0.998	0.994	0.979	0.936

TABLE V: Impact of the variation of the offered bandwidth in the network on the Fairness index: case of ring topology of 21 nodes

Fi experiences a slight decrease. Additionally, we remark that both of the protocols keep always very close values and in some scenarios, e.g., in case of 2mb/s and 5.5 mb/s, HsF-MAC slightly outperforms the standard BEB.

Fig. 4 shows the variation of the detection ratio of HsF-MAC in function of the percentage of cheaters in the network and the applied Misbehavior Coefficient (MC)[3]. As we can see from the plotted curves, the detection ratio reaches its highest values when the number of cheaters is low and the MC is high (the detection ratio is close to 100 % when the MC is equal to 0.8 and 1). Then it decreases as the percentage of cheaters increases, where we observe that the higher the MC is the more is the reduction of detection ratio. This decrease is justified by the rise of the collision rate among the cheating nodes and the interferences that prevent the receiver node from estimating the correct deviation of the monitored node.

The accuracy of a detection scheme is an important metric that evaluates its capabilities to distinguish the honest nodes from the cheaters. As shown in Fig. 5, no false alarm or missed detection occurred in HsF-MAC (detection accuracy is 100%) in the case of network characterized by a ring topology for different values of the MC. This is because of the absence of interferences that may disrupt the observations taken by the nodes to compute the backoff values of their neighbors. However, in the case of random topology the detection accuracy decreases slightly till it reaches its lowest value which is equal to 96% in the scenario of 150 nodes and MC equals to 1. We observe from this figure that the detection accuracy is inversely proportional to the network size and density. We remark also that the extent of the misbehavior level doesn't affect much the detection accuracy of HsF-MAC since the false alarms and mis-detected cheaters are caused by the interferences due to the hidden terminal phenomenon.

Despite the high detection rate of HsF-MAC, it still suffers from the false accusation of some honest nodes due to false alarms and miss at detecting some cheaters. To overcome this drawback, an adequate adjustment of the accuracy factor (discussed in section II-A) based on the network density, the interference level and the collision rate in the node's neighborhood remains a good solution to alleviate the impact of these false alarms and the mis-detected cheaters missed from HsF-MAC.

[3]The MC is a metric that measures the misbehaving level of a cheater node (i.e., it defines how much the cheater is deviating from the backoff value that it should wait for before accessing the medium).

IV. Conclusion

We have designed a novel backoff scheme to quickly detect the cheating nodes that do not comply with IEEE 802.11 MAC protocol. The major advantage of HsF-MAC over the existing solutions is its ability to reveal the identity of the node that doesn't choose its backoff properly after one successful transmission of an RTS frame. Therefore, the impact of the cheater node on the bandwidth fair-share is counteracted efficiently. Moreover, HsF-MAC is resistant to sender-receiver collusion. The simulation results show that the gain of this solution is twofold, it ensures fair share of bandwidth among nearby nodes, as shown in Tables III and V, and achieves high detection rate and low false positive rate, under moderate percentage of cheating nodes.

In addition, a reaction scheme is proposed to penalize the detected cheaters. Unlike the traditional punishment schemes, our reaction mechanism encourages the cheaters to become honest rather than punishing them. This is achieved through the warning message disseminated by the detector node, which constitutes double notification. It warns the cheater node regarding its misbehavior and discloses its identity to its neighbors. So, the cheater node will either stop its misbehavior to avoid punishment or keep it until being punished by the neighbors.

V. Acknowledgement

"Supported, in part, by Science Foundation Ireland grant 10/CE/I1855".

References

[1] R. Jain, G. Babic, B. Nagendra and C. Lam, "Fairness, Call Establishment Latency and Other Performance Metrics", *Technical Report ATM_Forum/96-1173*, ATM Forum Document, Aug. 1996.

[2] "IEEE 802.11 wireless LAN media access control (MAC) and physical layer (PHY) specifications", ANSI/IEEE Std 802.11, 1999.

[3] V. Gupta et al., "Denial of service attacks at the MAC layer in wireless ad hoc networks", *In Proc. of Military Communication Conference (MILCOM'02)*, California, USA, Oct. 7-10, 2002.

[4] P. Kyasanur, and N. H. Vaidya,"Selfish MAC Layer misbehavior in Wireless Networks", *IEEE Transactions on Mobile Computing*, 4(5):502-516, Sep/Oct 2005.

[5] M. Raya et al., "DOMINO: Detecting MAC Layer Greedy Behavior in IEEE 802.11 Hotspots", *IEEE Transactions on Mobile Computing*, Vol. 5, No. 12, pp. 1691-1705, Dec. 2006.

[6] A. A. Cardenas et al., "Evaluation of Detection Algorithms for MAC Layer Misbehavior: Theory and Experiments", *IEEE/ACM Transactions on Networking*, Vol. 17, No. 2, pp. 605-617, Jul. 2008.

[7] C. Jaehyuk et al., "A Lightweight Passive Online Detection Method for Pinpointing Misbehavior in WLANs", *IEEE Transactions on Mobile Computing*, Vol. 10, No. 12, pp. 1681-1693, Oct. 2011.

[8] N. Jaggi et al., "Distributed Reaction Mechanisms to Prevent Selfish Misbehavior in Wireless Ad Hoc Networks", *In Proc. of IEEE Globecom 2011*, Houston, Texas, USA, 5-9 Dec. 2011.

[9] G. Lei, C.M. Assi and A. Benslimane, "Enhancing IEEE 802.11 Random Backoff in Selfish Environments", *IEEE Transactions on Vehicular Technology*, 57(3):1806-1822, May. 2008.

[10] S. Djahel and F. Naït-Abdesselam, "FLSAC: A New Scheme to Defend Against Greedy Behavior in Wireless Mesh Networks", *International Journal of Communication Systems (IJCS)*, 22(10):1245-1266, Jun. 2009.