Cross-layer attacks in wireless ad hoc networks [1]

Svetlana Radosavac, Nassir Benammar and John S. Baras
Electrical and Computer Engineering Department
and the Institute for Systems Research
University of Maryland
College Park, MD 20742
e-mail: {svetlana,nassir,baras}@isr.umd.edu

Abstract — **Denial of Service (DoS) attacks are difficult to prevent and protect against. In this paper we focus on DoS attacks in wireless ad hoc networks that propagate from MAC to routing layer, causing breaking of critical routes. We present several traffic patterns that an intelligent attacker can generate to cause Denial of Service attack in one or several nodes in ad hoc networks. More specifically, we focus on the properties of IEEE 802.11 MAC protocol and attack propagation to the routing layer. We focus on attacks that use low-rate traffic patterns with the goal of disabling one or more specific nodes or partitioning the network. We propose a scheme for attack detection based on modelling of MAC protocols using Extended Finite State Machines (EFSM) and present general outline for an Intrusion Detection System that has the ability to generate attack patterns and check the validity of communication patterns in the network.**

I. Introduction

A MANET is a collection of wireless mobile nodes that are capable of communicating with each other without the use of network infrastructure or any centralized administration [1]. In addition to the wide range of attacks that are similar to the ones performed in wired networks, mobility, limited bandwidth and limited battery life present opportunities for launching novel attacks. A new class of attacks, cross-layer attacks, emerges from lack of interaction between MAC and routing layers. These attacks propagate from the MAC layer, where they are manifested as Denial of Service (DoS) attacks, to the routing layer, causing serious degradation of network performance in terms of the achieved throughput, latency and connectivity. An attacker can cause congestion in the network by either generating an excessive amount of traffic [3, 4] or by generating specific traffic patterns that prevent certain nodes from communicating with other nodes.

In this paper we focus on DoS attacks performed by single and colluding malicious nodes in the MAC layer of ad hoc wireless networks. We address the attacks that employ legitimate communications which result in node misbehavior and attack propagation through the network. In our scenario single and colluding malicious nodes utilize legitimate communication patterns in the MAC layer to isolate one or multiple nodes in the network and break existing paths in the routing layer. By disabling nodes in his vicinity, the attacker increases the probability of including himself in the new routes. We assume that each of the attacks performed is a multi-stage attack and that the attacker has some prior knowledge about the network structure and the location of the nodes he wants to attack.

The existence of cross-layer attacks implies lack of interaction between MAC and routing layers. We emphasize the necessity of improving the performance of a particular MAC layer protocol by implementing cooperation with the routing layer and vice versa. The cross-layer interaction should provide information about traffic levels (or other parameters) at critical nodes in the network and cause reaction in appropriate layer when congestion is detected.

MAC protocols are easier to model than routing protocols and it is relatively easy to represent any MAC protocol with Extended Finite State Machine (EFSM) representation. Most attacks in the MAC layer can be represented as loops in the EFSM representation. Additional criteria such as timing constraints or critical parameter values need to be imposed in order to distinguish malicious actions from increased traffic and therefore decrease the number of false alarms.

The paper is organized as follows. In section II we provide a short overview of issues addressed in this area so far and in III we model IEEE 802.11 MAC layer using EFSMs. In IV we address already mentioned issues of cross-layer interaction and present several cross-layer attacks and their consequences. In V we present an outline of Intrusion Detection System that is based on theorem proving, where theorems represent series of rules that a fault-free MAC protocol cannot violate. Finally, in VI we present experimental results and in VII we discuss the results and present an outline for future work.

II. Literature overview

Very little work has been done in the area of cross-layer attacks. In [3] the authors study the interaction of the routing and MAC layer protocols under different mobility parameters. They simulate interaction between three MAC protocols (MACA, 802.11 and CSMA) and three routing protocols (AODV, DSR and LAR scheme) and perform statistical analysis in order to characterize the interaction between layers in terms of latency, throughput, number of packets received and long term fairness.

[1]This material is based upon work supported by, or in part by, the U.S. Army Research Laboratory and the U.S. Army Research Office under grant No DAAD19-01-1-0494

They conclude that it is not meaningful to speak about a MAC or a routing protocol in isolation. More work has been done in the area of MAC layer protocol analysis. In [4] the authors address the issue of DoS attacks in IEEE 802.11 MAC and FAIRMAC protocols. They consider scenarios where an attacker causes congestion in the network by either generating an excessive amount of traffic by itself or by having other nodes generate excessive amounts of traffic. FAIRMAC performed better than 802.11 MAC, but both protocols showed significant decrease in traffic rates when DoS was mounted. In [5] the authors address the problem of misbehaving nodes in the IEEE 802.11 MAC layer and propose changes to IEEE 802.11 that would mitigate the effects of node misbehavior in the network. However, the authors do not address the issue of colluding nodes or any other kind of misbehavior apart from malicious Contention Window updates.

III. MAC layer protocol representation

The IEEE 802.11 DCF protocol specifies a Distributed Coordination Function (DCF) which is based on the same RTS/CTS message exchange as in MACA/MACAW. Unlike in MACA/MACAW, a node in IEEE 802.11 DCF defers only until the end of CTS frame reception. This solves both the hidden and exposed node problem. The only points where it differs from MACA are in the avoidance of collisions before transmitting RTS and its requirement of ACK transmission by the receiver after the successful reception of the data packet. The scheme follows the exponential backoff algorithm.

MAC protocols are easier to manage and represent than routing protocols. The nature of MAC protocol interactions, where event ordering and correct timing have crucial roles impose the necessity of using ordered models of execution with explicit timings. Explicit timing needs to be introduced in the model of event ordering due to the nature of event interactions in the MAC protocol (for example to describe timeouts). In this paper we represent IEEE 802.11 protocol in the form of EFSMs. Following the approach taken in [6] and modelling of PCF protocol in [7] it is straightforward to represent 802.11 MAC layer protocol using EFSMs.

Transmissions in 802.11 MAC layer are separated by inter packet gaps known as Inter Frame Spaces (IFS). Channel access is granted based on different priority access. The DIFS (DCF IFS) is used by STAs operating under the DCF for frame transmission. A station using the DCF shall be allowed to transmit if it determines that the medium is idle after a correctly received frame, and its backoff time has expired. It has the lowest priority. SIFS is the shortest of the interframe spaces. It is used when the stations have seized the medium and need to keep it for the duration of the frame exchange sequence. Several timers need to be introduced in order to specify the exchange of messages between nodes i and j. We introduce T_{DIFS} (DIFS timer), T_B (backoff timer), T_{SIFS} (SIFS timer), T_{OUT}, a timer set to a predetermined value

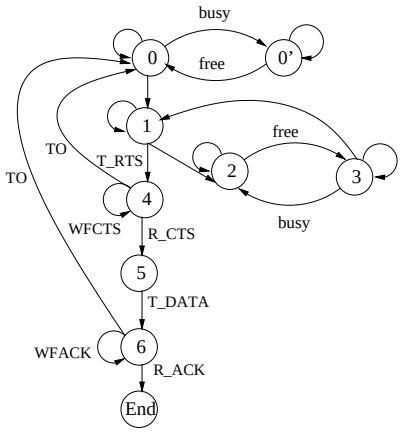

Figure 1: FSM representation of MAC protocol.

when a node is waiting for a reply. If the reply doesn't arrive during the specified period, timer is set into time out mode (it has expired) and the node makes transition into corresponding error state (or initial state). We also introduce $T_{RTS/CTS}$ that is set to a value that is defined in RTS/CTS message that the node overhears. This timer is activated when node makes a transition from state 0 to state 0' in figure 1.

All timers can be either active or inactive. Additionally, the timer can be expired (it's value has reached 0). EFSM representation of the node that is sending data is represented in figure 1.

In order to send data the node first needs to send RTS. This is represented as transition from 0 to 1. Transitions $1 \rightarrow 2$, $2 \rightarrow 1$, $2 \rightarrow 3$ and $3 \rightarrow 3$ represent part of the protocol previously described when the node waits for DIFS and backoff period to expire. When the medium becomes free and the timer is decremented to zero the node transmits RTS and makes a transition into the next state, where it waits for CTS (WFCTS) from node j. If CTS doesn't arrive during T_{OUT} the node makes a transition into state 1 and sets higher backoff period (maximal value is 256). Otherwise, if CTS is received, it makes a transition into state 5, waits for T_{SIFS} and transmits data. In state 6 it waits for ACK from node j. It makes a transition to either state 0 or state 1, depending on whether the ACK_j reaches node i or not. Transition from state 4 to state 1 represents the case when the destination node is either out of range or its CTS signal cannot reach the transmitter for some other reason. The case when multiple RTS signals collide and never reach the destination is also included in this transition since the transmitting node waits for CTS not knowing that RTS never reached the destination.

In case when node i hears RTS or CTS meant for node m, where $m \neq i$, it makes a transition to state 0', where it waits for $T_{RTS/CTS}$ and upon expiration it returns to state 0.

Finite State Machine representation of the node that

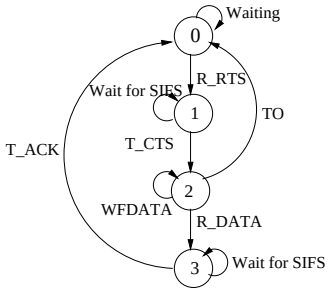

Figure 2: FSM representation of the receiving node.

is receiving data is represented in figure 2.

If node i receives RTS_{ji}, it waits for T_{SIFS} and transmits CTS_{ij}, makes a transition to state 2, where it waits for data from node j for T_{SIFS}. If the timer reaches 0, node i returns to the initial state. Otherwise, upon receiving data it makes transition to state 3, sends ACK to node j and returns to state 0. Due to space limitations we do not present the set of all predicates, transitions and events for receiving node and node k, where $k \neq i, j$, since it can be represented in similar fashion as the sending node.

IV. MAC layer issues in wireless networks and cross-layer interaction

As the results of [3] show, MAC and routing layers interact in numerous ways. Although the authors don't address malicious behavior of nodes, it is obvious that cross-layer interaction can be abused by malicious nodes to mount DoS attack in the MAC layer and propagate it to the routing layer. Contention at the MAC layer causes a routing protocol to respond by initiating new route queries. The same holds vice versa. Specific routes chosen by the routing protocol can significantly affect the performance of the underlying MAC protocols. This enables the intruder not only to break the existing routes, but also to maximize the probability of including himself in the new routes by maximizing the number of nodes he is disabling while minimizing the probability of being detected. In [5] the authors address the problem of selfish nodes, but the same scenario can be used by malicious nodes as well.

MAC layer has mechanisms to protect itself from congestions, but these mechanisms can be abused by attackers and used to disrupt communication in the MAC layer. Namely, the basic mechanism of MAC layer exchanges a series of control signals before it sends the data. If the control signals at either sender or receiver side are not received within a certain period of time, the signal is retransmitted (an upper bound on the number of transmission exists). All communication is done at the MAC level and there are no signals that are passed to the higher levels except the final ACK signal that notifies the routing layer that the data has been successfully forwarded to the next hop. However, the failure of service at the MAC layer causes route disruption at the routing level. As we will see in this section, the attacker can use the MAC layer properties to disable and isolate several key nodes and partition the network. Therefore, attack-resilient MAC protocol should have communication with both routing layer and Intrusion Detection System. When congestion is detected in either MAC or routing, the layer where the congestion originates should pass that information to the other layer and to the IDS. IDS should detect if the congestion is an attack and based on that decision the routing/MAC decide on future actions: to create new routes or discard the activity of the node that is causing congestion and pass that information to the other nodes. This implies that the system should monitor various parameters that are characteristic to MAC and routing protocols and based on their values make decisions about future actions. The general guidelines for parameters that can be exchanged between layers are given in [3].

It is obvious that several types of attacks can be performed in the MAC layer. First of all, an attacker can keep the channel busy so that the normal node cannot use it for transmissions, which leads to DoS attack in that node. The nodes follow binary exponential backoff scheme that favors the last winner amongst the competing nodes. This leads to the capture effect where nodes that are heavily loaded tend to capture the channel by continuously transmitting data which makes lightly loaded neighbors to back off continuously.

Based on the previous analysis, we classify a node as *normal* if it obeys the rules of MAC layer protocols when both sending and receiving packets. This type of nodes will not behave selfishly and will reply to RTS requests from other nodes and will update their CW, NAV etc. according to the rules of the protocol. A node is classified as *malicious* if it employs legitimate communication with other malicious or normal nodes which results in DoS in one or multiple nodes and attack propagation through the network. Finally, a node is classified as *misbehaving* if it denies to follow the rules of the protocol in order to gain priority in the network or disrupt already existing routes. This group of nodes includes wide range of behavior: from malicious nodes that start misbehaving after a certain point in order to maintain the priority up to nodes that jam the network with large number of packets. Misbehaving nodes can change the value of CW, NAV value, Duration/ID field in the packet etc.

A Cross-layer attacks

All attacks in this section include both malicious and misbehaving nodes. We use the realistic scenario, where each node initially employs legal communication patterns that prevent other nodes from communicating and after some time they start misbehaving in order to maintain priority in the network.

1. **Attack 1**
   Suppose that nodes A or D in figure 3 want to send

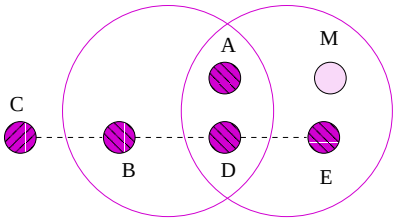

Figure 3: Attack scenario 1.

data. Denote the malicious node as M. Node M captures the medium before node A(D) decides to send data. Therefore, nodes A and D backoff for $T_{RTS/CTS}$. At the end of transmission node M will have to wait for t_{DIFS}+CW_{min} while A will wait fort_{DIFS}+CW, where $CW > CW_{min}$. When A tries to send a packet it will either sense the medium busy and stay in the same loop or it will eventually collide with RTS of node M. In this case its set of transitions is infinite loop. Node C, that wants to send a package through node D that is in the range of node M also cannot send any data. C sends a package to B, but B cannot receive any response from D because M has captured the medium. This attack addresses the unfairness of the 802.11 protocol since node that constantly fails to send data has worse chance to be enabled to send data as time passes. Hence, it is more likely that nodes with large CW that are backing off will be declared dead by other nodes than to get an opportunity to transmit.

As a consequence, the throughput of the system is degraded. To be able to detect this kind of malicious behavior, cooperation of MAC and routing layers is required.

2. **Attack 2**
   By investigating traffic the attacker can find out which routes have higher priority. In the second step, mounting an attack from the MAC layer an attacker congests the channels and breaks multiple routes, increasing the possibility that in the new route search it is included in the new path. The network layer part of the attack could increase the probability of the node being included in the new path by false route advertisements or some other method that would increase the probability of node being included in the path in case multiple paths are left. In case of attack 1, the route $C \rightarrow B \rightarrow D \rightarrow E$ will be broken and the new route will be $C \rightarrow B \rightarrow A \rightarrow M$.

3. **Attack 3**
   We are observing a system that contains 2 malicious nodes. Those nodes are not directly cooperating, they are out of range of each other, but both are in the range of the attacked node N. The

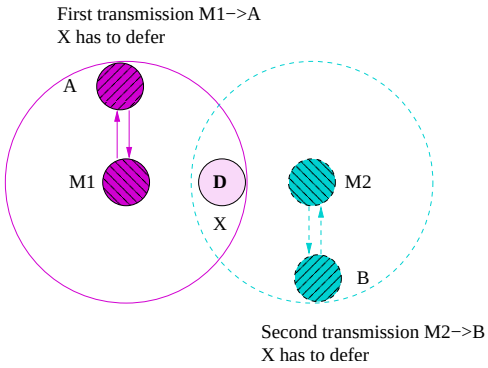

Figure 4: Attack scenario 5.

attack scenario can be performed as follows. Malicious node M_1 sends RTS to node A. RTS has information that the medium needs to be reserved for time t_1. At time t node N receives RTS from M_1 and defers its transmission for that period of time. Suppose that node M_2 needs to transfer data. It sends RTS to node B t_{DIFS} before the expiration of waiting period that was imposed by M_1's transmission. Node M_2 waits for t_{DIFS} and exactly at time when the first transmission stops this one starts and the medium is reserved. Since M_1, A and M_2, B are out of reach of each other but both can be heard by node N, they can continue their transmission infinitely many times unless additional fairness constraints aren't added. The described scenario is represented in figure 4.

V. Attack detection

One of the challenges we are facing in protecting an ad hoc network against DoS attacks, apart from distinguishing normal from abnormal traffic, is distinguishing congestions caused by malicious and non-malicious actions while minimizing the number of false alarms. As we have seen in previous sections, it is not meaningful to speak about neither MAC nor routing protocol in isolation. MAC layer protocols significantly influence routing protocols and vice versa. However, we have already mentioned that current interaction between MAC and routing protocols is limited to the exchange of ACK signals when the data is already sent. In order to mitigate the effects of congestion we need to design new dynamically adaptive protocols that can adapt to changing network and traffic characteristics by measuring and exchanging parameters that characterize cross-layer interaction and providing alternate routes with less traffic. However, in case of attacks that start in either the MAC or routing layer, providing alternate routes may represent an opportunity for the attacker to include himself in the new routes. Hence, when incorporating cross-layer interaction we need to include interaction with an Intrusion Detection System. In case when IDS relies only on measuring

traffic rates the number of false alarms rapidly increases. This implies the necessity of introducing a more complex system that would observe both traffic rates and several other protocol-related parameters, such as CW, NAV, injection rate, etc. and impose timing constraints. The MAC and routing layers would have to cooperate with each other in order to avoid points of congestion and reroute traffic and with the IDS in order to avoid inclusion of malicious nodes in the new routes or to isolate malicious nodes and propagate the information throughout the network.

For attack detection we use the proposed EFSM models of communicating nodes presented in III. Additional parameters are needed for determining the nature of transitions in order to avoid false alarms. Due to the fact that nodes in ad hoc networks cannot be perfectly synchronized due to mobility and other parameters, the nodes participating in attacks presented in IV start misbehaving after some time due to the fact that absolute synchronization that leads to blocking of targeted nodes cannot be maintained in real ad hoc network. In order to perform the attack without letting the attacked node(s) communicate, the attacker(s) need to violate one or more rules. As it can be seen from figure 1 there are several possible cases that lead to breaking of routes. Due to a high traffic rate a node may not be able to receive any requests from neighboring nodes. If it is already included in a route when the congestion starts, it will not be able to respond to any requests and eventually the connection times out and the route is broken. In this case the IDS should notify MAC and routing layers that there are no malicious nodes and that they are free to include any node in the new route. In case when the observed node is attacked, the attacker will have to change some parameters, i.e. CW size or the value of NAV in order to gain priority and stop the node from sending packets. The connection times out and the new RREQ is sent. The malicious node will maximize the probability of including itself in the new route by blocking as many nodes in its vicinity as possible while minimizing the probability of detection. In this case IDS should detect node misbehavior by controlling critical parameters that are exchanged in communication and marks node M as malicious.

As we can see, observing loops in the EFSM model provides information about possible sources of attacks, but in order to distinguish between attack and congestion additional parameters or timing constraints are needed. Therefore, we need to come up with the unified automatic approach for detection of wide range of attacks on wireless MAC protocols. In addition, there exists a need for creating a database of attacks that cover a significant range of attacks and is used as input for IDS.

For attack detection we formulate *theorems* that represent series of rules a fault-free MAC protocol cannot violate. Each property is formalized as a logical formula using temporal logic. We propose using Computational Tree Logic (CTL). For attack detection Automatic Model Checking is executed with input of the relevant rule (theorem) parameters from the nodes under examination. The general task is to check for a given CTL model $\mathcal{M}$, state $s \in S$ and CTL formula ϕ whether the property ϕ is valid in state s of model $\mathcal{M}$: $\mathcal{M}, s \models \phi$. For example, we define the rule that prohibits two processes to be in their critical section at the same time as:

$$AG(\neg(P_i.s = C \wedge P_j.s = C))$$

where $P_i.s$, $P_j.s$ represents states of a process and C represents a critical section. An important rule that excludes the appearance of infinite loops says that a process that wants to enter its critical section is eventually able to do so and is represented in CTL as:

$$AG(P_i.s = A \Rightarrow AF(P_i.s = C))$$

where A stands for an attempt. In case when the negated CTL formula is accepted by the EFSM, the safety of the system is endangered. The EFSM execution path is used for automatic attack generation. The model checker explores the search space for errors and generates a set of error scenarios that can later be used of protocol testing. We choose a specific set of parameters and in the case of an attack we save the parameters that differ from the normal values and add the specified set of parameters to the specific states in previously generated error scenarios. For that purpose we need to identify parameters that are used for error/attack detection. A useful extension is addition of invariant constraints that must hold in every reachable state of the observed model.

VI. Results

The experimental results do not incorporate any elements of cross-layer cooperation for now. We present the results of proposed attacks on IEEE 802.11 MAC. For better illustration of the above attacks, we have used OPNET to simulate the behavior of nodes under the attack.

The first scenario is presented in 3. We simulate network traffic with duration of 120 seconds. In the first half of the simulation, the malicious node is inactive and node D is able to send packets to its neighbor. After 70s, the malicious node starts attacking node D.

The second scenario is presented in 4. It is important to realize that the malicious node does not need to send the traffic to node D in order to disrupt its traffic. The main intention of node M is to broadcast the RTS packet to node D, so it updates its Network Allocation Vector and doesn't send any traffic for the communication period indicated in the duration field of the RTS packet.

This attack scenario requires synchronization between two malicious nodes M_1 and M_2. The nodes need to alternate while sending traffic and therefore they need to generate packets at half the rate of the previous scenario. The major advantage of this attack is that it is more difficult to detect. Figure 6 shows the data traffic sent by

the attacked node for 2 data traffic generation rates of the malicious nodes. In the first figure node D is still able to send its packets. However, when the attack is mounted, node D is completely disrupted during the 30s attacking period.

VII. Discussion

We can now observe that if there is no cross-layer cooperation or IDS, the malicious nodes can include themselves in the new routes and continue the attack in the routing layer. Hence, significant cross-layer cooperation is needed in order to mitigate the effects of congestion. In addition, cooperation with IDS is needed in order to avoid malicious nodes and to inform the rest of the network of their existence. An efficient IDS should find violations in communication patterns in the MAC protocol and automatically generate a database of attacks by saving the time history together with values of related variables. When the MAC protocol is under attack, the model checker needs to prove that the behavioral pattern that the protocol has is indeed malicious and using the cross-layer interaction alternate routes that are not under the attack are provided and the information about

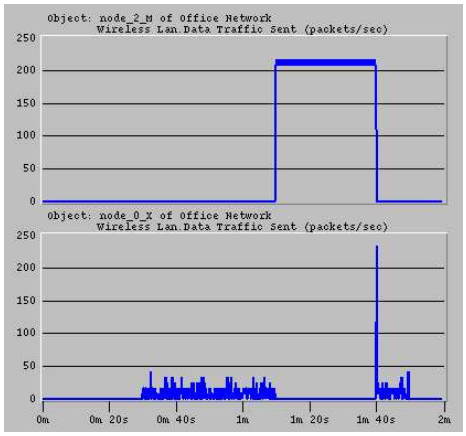

Figure 5: Data Traffic sent by nodes M and D.

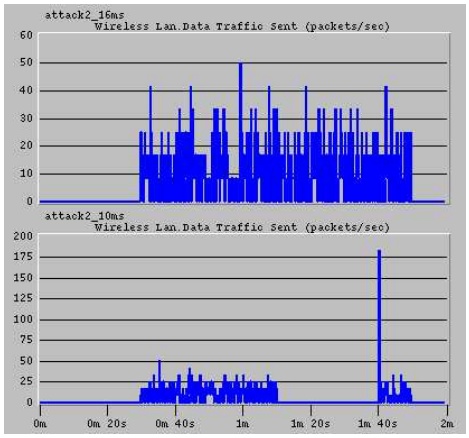

Figure 6: Data Traffic sent by node D.

the malicious nodes is passed to the higher layers. The same holds vice versa. When there is an ongoing attack in the routing layer, the system needs to make alterations in the MAC layer communication to be able to provide new routes that are free of intrusions.

In the next stage of our work we plan to extend theorems for violation of MAC protocol rules and examine parameters that need to be exchanged among MAC and routing layers. Given that event ordering and correct timing have crucial roles in MAC protocols we plan to introduce explicit timing constraints in our safety properties definitions. As in all model checking approaches, state space explosion represents a problem during transformation from a CTL formula to an FSM. A potential approach is a combination of model checking and theorem proving techniques.

Using CW and NAV violations we plan to generate our own database of attack features using automatic attack generation capability of EFSM model.

References

[1] H. Deng, W. Li, D. P. Agrawal, *Routing Security in Wireless Ad Hoc Networks*, IEEE Comm. Magazine, October 2002.

[2] S. Marti, T. J. Giuli, K. Lai and M. Baker, *Mitigating Routing Misbehavior in Mobile Ad Hoc Networks*, MOBICOM 2000, Boston, MA

[3] C. Barrett, M. Drozda, A. Marathe, M. V. Marathe *Analyzing Interaction between network protocols, topology and traffic in wireless radio networks*, Proc. IEEE Wireless Comm. and Networking Conference (WCNC'03), New Orleans, Louisiana, 2003.

[4] V. Gupta, S. Krishnamurthy, M. Faloutsos *Denial of Service Attacks at the MAC Layer in Wireless Ad Hoc Networks*, IEEE Milcom 2002, Anaheim, California, October 7-10, 2002.

[5] P. Kyasanur, N. . H. Vaidya *Diagnosing and Penalizing MAC Layer Misbehavior in Wireless Networks*, Technical Report, December 2002.

[6] A. Helmy, S. Gupta *STRESS: Systematic Testing of Protocol Robustness by Evaluation of Synthesized Scenarios*

[7] M. A. Youssef, R. E. Miller *Analyzing the Point Coordination Function of the IEEE 802.11 WLAN Protocol using a systems of Communicating Machines Specification*, UMIACS Technical Report 2002-36