**CERIAS Tech Report 2005-61**

# CHANNEL ACCESS AND SYNCHRONIZATION ATTACKS AGAINST MAC PROTOCOLS IN WIRELESS NETWORKS

by Gunjan Khanna, Ammar Masood, Cristina Nita Rotaru

Center for Education and Research in
Information Assurance and Security,
Purdue University, West Lafayette, IN 47907-2086

# Channel Access and Synchronization Attacks Against MAC Protocols in Wireless Networks

*Abstract*— **The 802.11 standard specifies mechanisms for channel access, data delivery, authentication and privacy for wireless communication. The standard makes no provisions for faulty, selfish or malicious behavior assuming that nodes always act according to the specifications of the protocol. Thus, nodes running defective protocol implementations, misconfigured, or compromised can potentially cause significant disruption in the network. In this paper we present an analysis of channel access denial of service attacks against 802.11b. We demonstrate the attacks through simulation and analyze them by considering the effect of multiple attackers, their relative positioning and the timing of the attack, missing from the previous papers. Our study of the attacks under varying timing provides a new direction for developing mitigation techniques at a higher layer completely independent of the Medium Access Control(MAC). In addition, we identify and describe new attacks against the beacon-based synchronization mechanism used for channel access and by the power saving mode in 802.11a, b, and g. We provide simulation results that demonstrate their feasibility and analyze them considering the attacker's effort versus the induced damage and effect on other protocols and services. Finally, we discuss some detection and mitigation techniques for the analyzed attacks, demonstrating the efficacy of several of them through simulations.**

## I. Introduction

The 802.11 [1] standard specifies a family of protocols developed by IEEE for wireless LANs. The most well-known, 802.11b [2], supports 11 Mbps in the 2.4 GHz band and specifies mechanisms for channel access, data delivery, authentication and privacy. Two operation modes are defined: *infrastructure*, in which nodes communicate with each other through an access point (AP) and *ad hoc* in which nodes communicate directly with each other without the use of an AP. In the first case, the standard also specifies the distribution services which enable a node to roam between several APs. Wireless communication protocols design must address major challenges not encountered in the wired environment. They include: 1) the difficulty to detect collisions; 2) the shared medium; 3) the limited available spectrum; and 4) the limited energy available to wireless devices. Thus, wireless protocols have to address the multi-path fading of a radio signal that may result in inability of detecting collisions, or the hidden terminal problem where transmissions from two or more senders – who do not hear each other's transmissions – may collide at a receiver. In addition, they must minimize the control information and maximize the utilization of the available spectrum, co-ordinate the access to the shared medium, and use energy efficiently by limiting not only transmissions, but also unnecessary listening of the channel. IEEE 802.11 copes with such challenges by using:

- two channel access mechanisms that provide collision avoidance: the Distributed Co-ordination Function (DCF), and the Point Co-ordination Function (PCF).
- a Power Saving Mode (PSM) that allows nodes to go to sleep while they are waiting for the channel to become available for transmissions. PSM requires synchronization between a point co-ordinator (PC) and the other nodes. The synchronization is achieved using beacon packets sent periodically by the PC.

In addition, 801.11 defines authentication services: Open System (a null authentication scheme) and Shared Key [1] where authentication is carried based on a shared secret key and is defined by the wired equivalent privacy (WEP) protocol. The WEP protocol also defines confidentiality and message integrity. Confidentiality is provided using encryption via a stream cipher, RC4, (relying on a secret key and an input vector sent in clear), while integrity is provided using the CRC32 scheme. 802.11 also provides several services that allow a node to move between several APs. The most important are the association and disassociation services which allow a node to inform an AP that it needs or no longer requires the service of that AP.

The standard makes no provisions for faulty, selfish or malicious behavior assuming that nodes always act according to the specifications. Thus, nodes running defective protocol implementations, misconfigured, or compromised can potentially cause major disruption in the network. Significant work has focused on analyzing security vulnerabilities in wireless networks at higher network layers, especially routing [3]–[11]. There is some work [12]–[14], [27] on studying the impact of selfish or malicious behavior at the MAC level, but has mainly focused on identifying attacks and not studying its impact in varying conditions like timing of attack, number of attackers and their relative positions. The absence of security mechanisms enforcing the correct functionality of the MAC layer can prevent many protection schemes proposed at higher levels to achieve their design goals.

In this paper we analyze the impact of denial of service (DoS) attacks against MAC protocols for wireless networks focusing on 802.11. More precisely, we use simulations to provide a detailed analysis of 802.11b channel access attacks introduced in [13], [14]. Our contributions in this paper are to 1) study the impact of multiple attackers, their relative positioning and the effect of timing of the attack, 2) significant contribution stems from the study of timing of attacks hinting at suitable higher layer protocols for preventing attacks at lower layers, 3) we identify a class of new attacks called

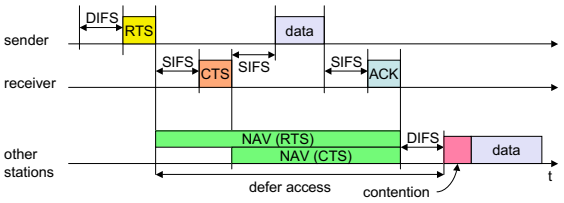



Fig. 1. Using NAV for channel reservation

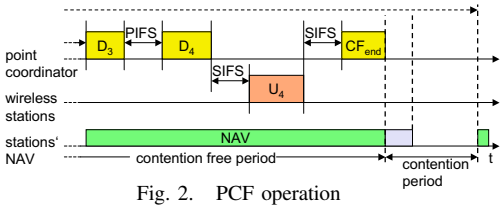



Fig. 2. PCF operation

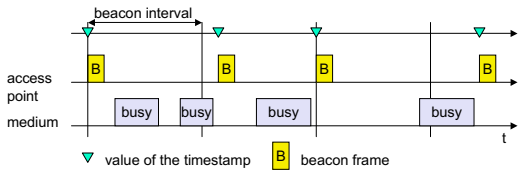



Fig. 3. Synchronization using beacon: infrastructure mode

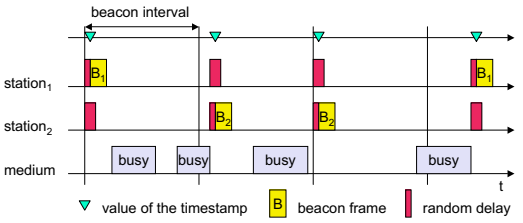



Fig. 4. Synchronization using beacon: ad hoc mode

*synchronization attacks* discussing their impact against the beacon-based synchronization mechanism used by the PCF and PSM components of 801.11b, and finally 4) we propose detection and mitigation techniques to counter each attack. The results for synchronization attacks indicate that while being low rate attacks, they bring the throughput down to near zero for PCF and can disrupt any service or protocol relying on synchronization (examples include protocols for sensor [17], and mesh [18] networks).

The rest of the article is structured as follows. Section I-A describes in details channel access mechanism of 802.11b, while Section I-B overviews prior research identifying vulnerabilities against 802.11. We describe the channel access and synchronization attacks in Section II, analyze them in Section III, and propose mitigation techniques in Section IV. Finally, we conclude the paper in Section V.

### *A. 802.11 Channel Access Mechanisms*

The core of the 802.11 standard are the mechanisms for channel access: the Distributed Co-ordination Function (DCF), which is mandatory and the Point Co-ordination Function (PCF) which is optional. DCF is a Carrier Sense Multiple Access/Collision Avoidance (CSMA/CA) protocol, which employs an exponential back off mechanism and a channel reservation scheme (presented in Figure 1). The purpose of the channel reservation scheme is to avoid the hidden terminal problem. A node desiring to send makes a request by sending a RTS packet, while the node receiving the data accepts the transmission by sending a CTS packet. The sender also specifies the time needed to transmit the data, through a Network Allocation Vector (NAV) value carried by both RTS and CTS packets. Nodes which lie within the listening range of a transmitting node record that NAV value and increase their back off time accordingly, since the channel will be busy.

PCF is a contention-free protocol, enabling nodes to transmit data synchronously, with regular time delays between data transmissions. This is achieved by using a point co-ordinator (PC) that controls which nodes can transmit during any given period of time. Within a time period referred as the contention free period, the PC polls all stations operating in PCF mode to check if they have data to transmit. Each node can transmit during a specific period of time. 802.11 allows DCF and PCF to coexist, defining how nodes alternate between DCF and PCF: on one period of time nodes will restrain from using CSMA, while the next time period, the nodes will wait for a poll from the PC before sending data frames.

In addition, 802.11 specifies how nodes can save power by using a Power Saving Mode (PSM). PSM specifies that a node can periodically go to sleep if it has no data to receive, and check with the PC (every time it wakes up) if data is waiting for it. A node can communicate to the AP its sleep schedule and the AP will buffer packets while the node sleeps.

Both PCF and PSM require synchronization amongst the PC and the stations. In order to achieve this synchronization, the PC transmits beacons at periodic intervals. The beacon packets are broadcasted in clear containing a time stamp. The nodes listening to the beacon adjust their clocks according to the time stamp in the beacon packet. Typically, the base station broadcasts a beacon frame periodically (10 to 100 times per second). In the PCF mode the initial beacon sent out by the PC denotes the start of the polling period or contention free period. As shown in Figure 2 the beacon contains a NAV indicating the length of the contention free period. All nodes back off according to this NAV value and wait for the polling packet for transmitting data. Each node on receiving the poll message transfers its data. No RTS/CTS mechanism is employed during the contention free period.

The beacon packet also contains the next time when a beacon will be transmitted. If the channel is busy, the beacon is transmitted when the channel becomes free and the time stamp is adjusted accordingly (see Figure 3). In case of an ad-hoc mode where no PC is available, synchronization using beacons is carried out in a distributed fashion (see Figure 4). At the start of each beacon interval, each node chooses a random back off timer and listens to the channel. If a beacon is heard before one's timer expires, then that node ends its timer and does not send a beacon. All nodes which hear the beacon adjust their clocks according to the time stamp value. If no beacon is heard and the timer expires then the node sends a

beacon containing the time stamp.

### B. Related Work

Several vulnerabilities have been identified for 802.11b. They include attacks against confidentiality and integrity, de-authentication and de-association attacks, and selfish behavior from nodes who obtain unfair access to the channel.

*Confidentiality and integrity*: Significant research focused on analyzing the confidentiality and integrity services provided by WEP for 802.11. Borisov *et. al.* [12] have shown several attacks against WEP that allow for modification of existing frames, injection of spoofed frames, as well as decryption of the communication, without breaking the shared secret key. The attacks rely on the reuse of the keystream of the encryption cipher and of the fact that integrity does not rely on a secret key. A total break of security of WEP, by recovering the shared secret key, was shown in [19]. Follow-up protocols such as Wi-Fi Protected Access (WPA) and 802.11i [20] were proposed to provide enhanced MAC layer security. For a detailed description of their limitations, see [21].

*Authentication and association*: An attack in which the authentication service can be easily bypassed, without knowing the shared secret WEP key was shown in [12]. DoS attacks against both authentication and association were shown in [22]. The authors proposed a new authentication framework to address the identified vulnerabilities. Both the authentication and association protocols suffer from a vulnerability in which a node or AP can explicitly request de-authentication or de-association from each other via a message that is transmitted in clear and is not authenticated. This creates opportunities for DoS, as it was successfully demonstrated in [13]. Finally, to provide a long-term solution to the security problems of 802.11, the 802.11 TGi working group has proposed the standard use of the 802.1X and 802.11i protocol. Recently a security analysis [21] of 802.11i pointed out several vulnerabilities and suggested improvements.

*Selfish use of the channel*: Several attacks were pointed out in which selfish nodes try to obtain unfair use of the channel by exploiting either the CSMA mechanism or the NAV-based channel reservation mechanism. [23] identified a number of security vulnerabilities in the MAC protocol, including attacks against the virtual carrier-sense mechanisms, but presented no empirical validation. [24] examined DoS MAC attacks in ad hoc networks and demonstrated that MAC fairness can mitigate the problem. The emulated fair MAC they experiment with, serves as a proof of concept but cannot be implemented in a distributed manner. Authors in [27] propose DOMINO to prevent greedy behavior in 802.11. They do not analyze the attacks with timing variations or number of attackers. A CSMA/CA back off based DoS attack against 802.11b was identified in [14]. The authors propose as defense to have the receiver controlling the back off window of the sender. An attack that exploits the channel reservation mechanism by setting the NAV value always to the allowed maximum, was presented in [13]. In addition, several mitigation techniques that do not use cryptographic protocols, were also proposed. We discuss the limitations of these mitigation techniques in Section IV.

## II. Channel Access Attacks Against 802.11b

In this section we present several attacks that are the focus of this paper. The attacks exploit the channel access mechanisms of the 802.11b MAC protocol. The NAV and CSMA/CA attacks were previously reported in [13] and [14], but provide an overview here to make their analysis in Section III easy to follow. The synchronization attacks based on beacon are new attacks on MAC of wireless networks.

### A. NAV-Based Attack

802.11b uses a RTS/CTS mechanism to avoid collisions due to hidden nodes. Nodes transmit a NAV value in the packets in order to indicate the duration of the transmission. All nodes who listen to the packet will back off for the time indicated by the NAV value.

An attack against the availability of the channel, exploiting the channel reservation mechanism is possible as follows. A malicious node, instead of behaving correctly, will send a RTS packet with a high NAV value. All the other nodes hearing the RTS (or the corresponding CTS) will mark this NAV value and back off accordingly. If the attacker keeps sending RTS packets with a high NAV then it can block the channel and hence cause a DoS in the network. The highest allowed value for the NAV is 32767 which is approximately 32 ms. The attacker can send 30 packets/sec each with the NAV set to 32767 making the channel access exclusive to itself. The other nodes constantly back-off as they find the channel busy, bringing their throughput potentially to zero.

### B. CSMA/CA-Based Attack

802.11b uses carrier sense multiple access with collision avoidance (CSMA/CA) to avoid collisions in the shared medium. Briefly, the mechanism works as follows: each node listens to the channel and sends a packet only if the channel is idle. If the channel is busy, the node will defer its transmission till the channel becomes available, plus an additional *contention* period to avoid collisions. Each node maintains a contention window divided in slots. The contention period is a randomly chosen slot based on the contention window size.

The CSMA/CA attack exploits the contention period. More specifically, if a (potentially compromised) node selfishly reduces its contention window, it will wait a smaller back off period and thus get priority in accessing the channel. This behavior can potentially lead to a decreased throughput for the other nodes in the network, because they will find the channel busy and follow exponential back off procedure. Thus a node with the smallest contention period will obtain the highest throughput, at the expense of the other nodes.

### C. Beacon-Based PCF Attack

As described previously in Section I-A, a beacon packet is used in the PCF mode to allow polling by the point coordinator (PC). A node or AP can act as the PC. The PC sends

the beacon packet to specify the start of the contention free period (CFP). The nodes go to sleep once they hear the beacon packet. The PC polls each node for data packets to transmit and once a node receives a poll message they discontinue their wait and send the data to the PC. During the CFP no RTS/CTS packets are transmitted. Instead a node transmits only when it receives the poll message. A malicious node can affect the throughput of other nodes by sending a beacon packet and holding back the poll message. In this case, the normal node will go to sleep while waiting for the poll message that never arrives and thus prevents it from sending data. The result can be a zero throughput for that node.

### *D. Beacon-Based Synchronization Attack*

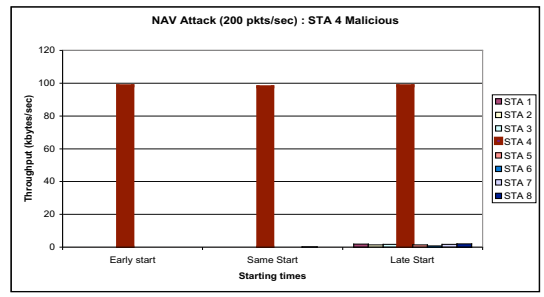
Both PCF and PSM require synchronization amongst the PC and the nodes, achieved via beacon packets. An important information carried by the beacon is the next time when a beacon will be transmitted, i.e. the start of the next CFP, along with the current time stamp. If the channel is busy during that time, the beacon is transmitted when the channel becomes available and the time stamp is adjusted accordingly. In the case of the ad-hoc mode where no PC is available, beacon-based synchronization is carried in a distributed fashion. At the start of a beacon interval each node chooses a random back off timer and listens to the channel. If a beacon is heard by a node before its timer expires then that node ends its timer and a beacon is not sent until the next beacon interval. All nodes which hear the beacon adjust their local clocks according to the time stamp in the beacon. If no beacon is heard and the timer expires at a node then it sends a beacon containing the current time stamp, if the channel is free.

PSM is used by nodes to save power while they are waiting for the channel to become available for transmission. A node can go to PSM i.e. sleep at a particular time which is supported by the AP. During the period of sleep the AP buffers the packets and hands them over to the node when it wakes up. If a node wakes up at a different time than what the AP expects, it can loose the data waiting for it. The result can potentially be a reduced throughput for the de-synchronized node.

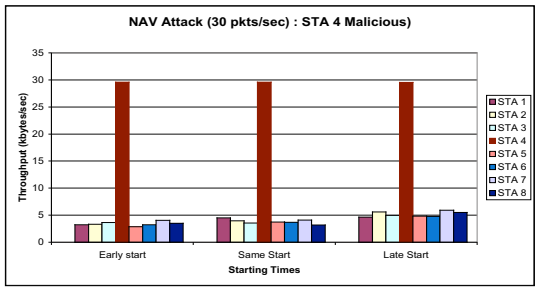
A malicious node can attack both PCF and PSM by de-synchronizing the clocks of correct nodes. The clock of a correct node can be deviated by sending a single beacon with a malicious time stamp value. This clock error will exist until a correct beacon is received by that node. The de-synchronization introduced will inherently de-stabilize any protocol or service which depends on synchronization.

## III. Simulation Results

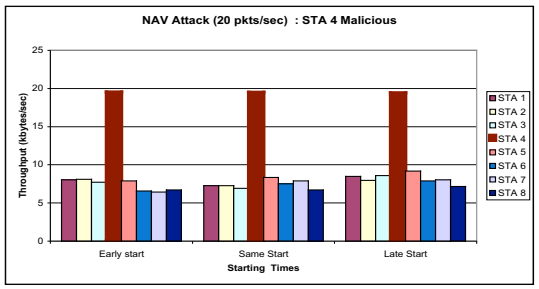
In this section we demonstrate the above described attacks through simulations and analyze their potential impact. We use the 2.27 NS-2 simulator [25] to simulate the attack scenarios. In addition, we implement a modified 802.11 MAC protocol that has the behavior of a malicious node. For both the NAV and CSMA/CA back off attacks, the simulation is conducted with a topology consisting of eight static nodes positioned around one AP, which is centrally placed amongst the nodes.




(a)




(b)




(c)

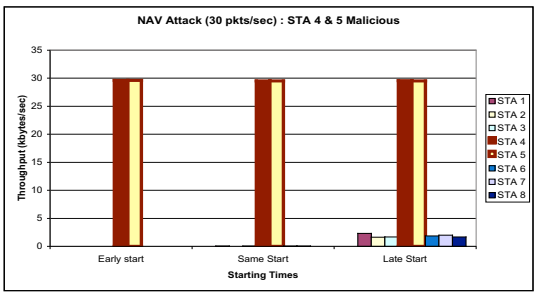
Fig. 5. NAV Attack: CBR flows, one malicious node

The nodes are not placed in a pure circular fashion around the AP, but following a rectangle, such that some nodes are closer to the AP than others to study the impact of location on the attacks.

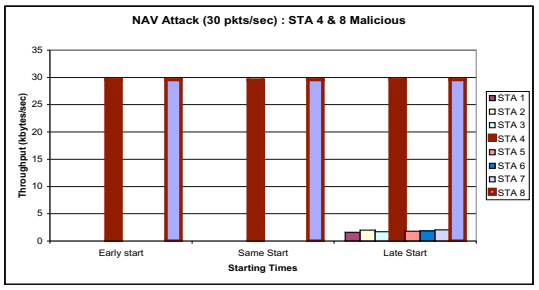
### *A. NAV-Based Attack*

We study the NAV attack using multiple attackers and varying the number of nodes, as well as the effect of the the transport mechanism on the attack. The attack is simulated by setting the malicious node duration value for RTS and CTS packets to the maximum possible value i.e. 32767. We use a topology of 8 nodes with a centrally placed AP. In order to study the effect of timing on attacks, We consider different starting times of malicious nodes as compared to the correct nodes. The traffic generating agent for "early start" of the malicious node is started 1 second earlier than other nodes whereas for "late start" it starts 10 seconds after other nodes. For all the scenarios the packet size is set at 1000 bytes and the simulation is run for 200 seconds.

*1) CBR Flows:* The CBR flows are established between all the nodes and the AP, the underlying traffic being provided by UDP agents attached to all the nodes.We investigate the impact of the NAV attack for CBR flows by selecting different inter

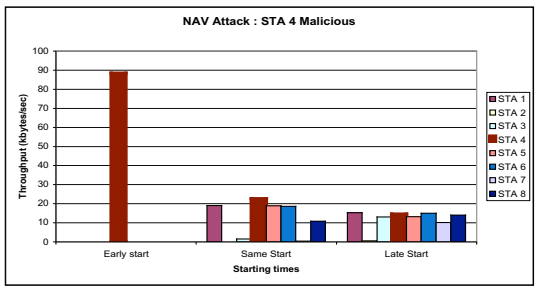



(a)

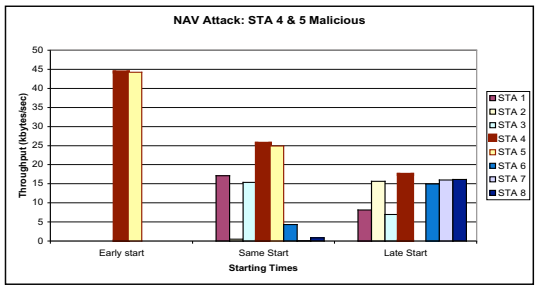



(b)

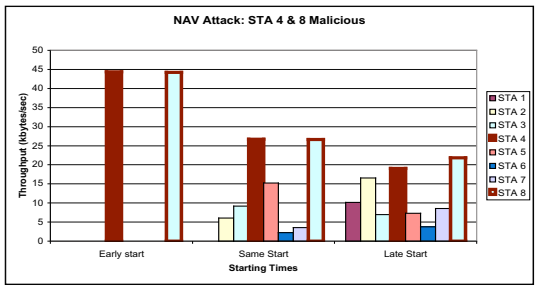
Fig. 6. NAV Attack: CBR flows and 2 malicious nodes




(a)




(b)




(c)

Fig. 7. NAV Attacks: TCP Flows

packet arrival times. The inter packet interval is set to 0.005, 0.033, and 0.05 seconds which corresponds to packet rate of 200, 30, and 20 pkts/sec respectively.

Figure 5 presents results when there is one malicious node, STA 4. As it can be seen in Figure 5 (a), for a CBR inter packet interval of 0.005 seconds (packet rate of 200 pkts/sec), for all the starting cases the malicious node is able to have full control of the medium. This is because all the other nodes hold their transmission by setting up their back off values to the duration values set in the malicious node's packets. The same behavior can be observed in Figure 5 (b), which indicates that just by transmitting 30 packets/sec the malicious node is able to effectively cause DoS for all the other nodes. This effect is prevalent for all the starting cases as the malicious node is able to obtain full control of the medium. Even at a low packet rate of 20 pkts/sec (Figure 5 (c)) the malicious node manages good control of the channel, but in this scenario other nodes also manage to transmit. The low packet rate of the malicious node enables other nodes to transmit after the time when the NAV duration limit set by the malicious node expires. For very low packet rates the malicious node is not able to affect the throughput of other nodes. This is because of the fact that whenever other nodes have a packet to transmit, they are not restricted by the duration value transmitted in the last packet sent by the malicious node because of larger inter packet interval of the malicious node.

Figure 6 investigates the effect of the placement of the malicious nodes. There are two malicious nodes in the network. In the first case the nodes are situated far apart (STA 4 and 5), while in the second case the nodes are adjacent (STA 4 and 8). The CBR inter packet interval is set at 0.033 (packet rate of 30 pkts/sec) since this was the minimum interval that was observed to cause DoS when only one malicious node existed in the network. As seen in Figure 6, the addition of another malicious node results into stronger control of the medium by the malicious nodes and a more pronounced DoS effect for the correct nodes. The starting times for the malicious node does not degrade the effectiveness of the attack. By comparing results from Figure 6 (a) and (b) it is evident that the relative location of malicious node does not affect the effectivness of the DoS attack for CBR flows.

Since NAV attack requires an attacker to initially contend for the channel and the attack is only successful once it gains the channel, higher layer protocol plays an important role in holding the channel. To further investigate this issue we repeat the experiments by replacing CBR flows with TCP flows because in TCP one has to relinquish the channel to receive the TCP Ack.

*2) TCP Flows:* Here we present results when traffic consists of TCP flows, established in between all the nodes and the AP. The underlying traffic is provided by TCP agents attached to all the nodes.

Figure 7 (a) depicts the throughput when only one node, STA 4, is malicious. It can be noticed that when the malicious node starts the TCP flow earlier than other nodes, it is able to take full control of the medium for the whole run of the

simulation. This results into a complete DoS for all the other nodes. However, for the "same start" and "late start" cases, the malicious node does not control the channel or in other words the attack is not successful. If a malicious node starts early then it gets exclusive access to the channel while in case of "same start" and "late start", the malicious node has to contend for the channel before it can cause DoS. The attack gets further mitigated when TCP is used because of the transmission of the TCP-Ack packet by the AP. Thus, whenever AP takes the channel away from the malicious node to transmit TCP-Ack packets it attempts to transmit such packets to the well behaved nodes as well. This takes the control of the medium away from the malicious node and as a result the impact is not as strong as for "early start" case where AP always transmits TCP-Acks packets to malicious node only, allowing it to control the channel.

This corroborates that a NAV attack to be successful needs support from higher layer to hold the channel for indefinite periods like in CBR. A suitably designed higher layer protocol which prevents holding longer one-way traffic sessions can mitigate NAV attack. To study multiple attackers, We introduce one additional malicious node in the network. Figure 7 (b) and (c) depict the case when there are two malicious nodes, again in the first case the nodes are far away, while in the second case they are adjacent. It can be noticed that for "early start" the malicious nodes cause complete DoS for all the other nodes. This does not happen however, for the "same start" scenario. Interestingly, in case of "late start" the throughput of one of the malicious nodes (STA 5) drops drastically to nearly zero whereas STA 4 is still able to maintain higher throughput than others. This effect is attributed to the combination of TCP acknowledgement aspect and the randomness in the selection of the congestion window by contending nodes. As seen in Figure 7 (c), the relative location helps the malicious nodes to maintain better control of the medium also for "same start" and "late start" cases.

### B. CSMA/CA-Based Attack

We simulate the behavior of a malicious node using the CSMA/CA back off attack, by varying the contention window (CW) size. The CW is presented as a fraction of the window used by the attacker over the actual window size. For example if the total slot size for a normal node is 32, and the CW fraction is 0.75, the attacker is using only 0.75*32 slots for calculation of the back off. Since the total slot size is reduced he choses a lower back off on average.

To analyze the effect of higher layer protocol we use both CBR and TCP flows. We vary the number of malicious nodes to identify the impact of the number of malicious nodes and location on the throughput degradation. The throughput is calculated for all nodes and plotted against varying values of the congestion window of the malicious node.

*1) CBR Flows:* The inter packet interval is set to 0.033 giving a packet rate of 30 pkts/sec.

Figure 8 (a) shows the throughput when there is only one malicious node (STA 4), whereas the rest of the nodes use

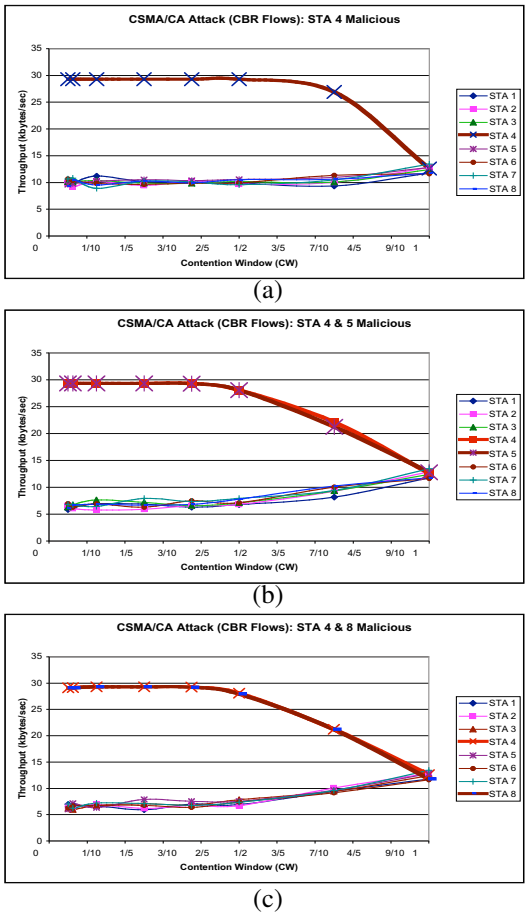
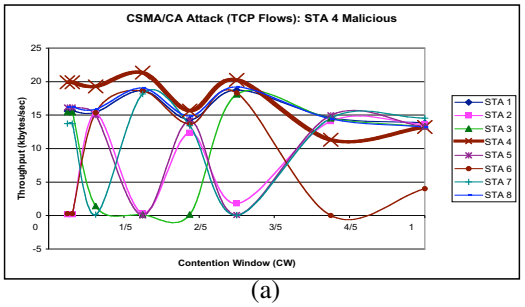



Fig. 8. CSMA/CA Attack: CBR Flows

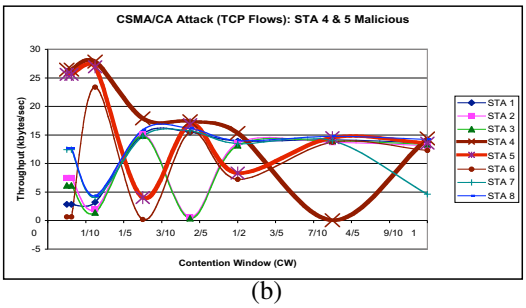
the standard 802.11b protocol. The results indicate that the malicious node maintains a higher throughput for all CW fractions less than 1. Even when the malicious node reduces its CW to 0.75, the impact is quite strong. For smaller values the effect is much more pronounced as the malicious node is able to take full control of the medium and maintain a very high throughput at the expense of the other nodes.

Figures 8 (b) and (c), show results for the case when two nodes are malicious. The nodes are chosen such that in first case (Figure 8 (b)), the malicious nodes (STA 4 and 5) are far apart from each other, while in the second case (Figure 8 (c)), the malicious nodes (STA 4 and 8) are close to each other. The experiment demonstrates that the malicious nodes maintain a very high throughput while sharing the medium, at the expense of a significant throughput decrease for correct nodes. For a CW of 0.5 or smaller, the two nodes control the channel. Also, by examining all three graphs from Figure 8 it appears that the relative location of malicious nodes does not affect the throughput degradation for the other nodes, in both cases, the malicious nodes being able to send at the maximum rate. This is also because the SNR observed by the malicious nodes is similar because of the CBR flow operating in the network which offers little or no variability.

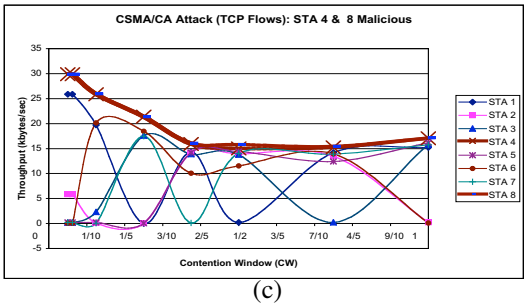
*2) TCP Flows:* We perform the same experiments for TCP flows established between all the nodes and the AP.




(a)




(b)




(c)

Fig. 9. CSMA/CA Attack: TCP Flows

Figure 9 (a) presents the throughput achieved, when only one node, STA 4, is malicious. The results indicate that the malicious node obtains a higher throughput than the other nodes for most of the CW values.

The graphs obtained in TCP differ distinctly from those in CBR. This is due to the fact that TCP traffic is two-way: data and acknowledgments (TCP-Ack) packets. The higher layer protocol (i.e. TCP) forces the MAC layer to wait for the TCP-Ack packet, hence relinquishing the channel. After the TCP-Ack is received, the attacker has to again contend for the channel by choosing a random back off. Because the attacker has to contend for the channel repeatedly and chose a random back off, the effect of the attack is not as pronounced as in CBR. In contrast, because of the one way flow in CBR, the attacker gets the channel repeatedly and is never relinquished. A malicious node may also corrupt the TCP functionality, but here we would like to draw attention to the coupling between layers which can be used to design secure protocols.

Another interesting observation is that the impact of the malicious node is different on the other nodes for different settings of CW. Figure 9 (a) shows that the malicious node (STA 4) reduces the throughput of nodes STA 2, 5 and 7 to nearly zero for its CW fraction of 0.5. However, at a CW fraction of 0.75 the impact is more pronounced for node STA 6. Since the channel capacity is fixed, a malicious node it will

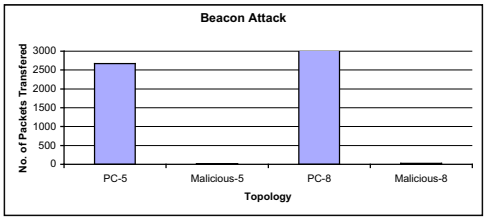



Fig. 10. Beacon-Based PCF Attacks: More Point-Co-ordinators

obtain more bandwidth at the expense of some other node(s).

The results presented in Figure 9 (b) and (c), consider scenarios when two nodes are malicious. In the first case (Figure 9 (b)), the malicious nodes (STA 4 and 5) are situated far from each other. When compared to their impact for CBR flows the results for TCP flows do not depict high throughput for the malicious nodes for all the CW fractions less than 1. For example, at a CW fraction of 0.75 the throughput of STA 4 nose dives to zero whereas STA 5 is still able to maintain a high value. However, the results do depict that for very low values of CW both malicious nodes are able to obtain full control of the channel. The difference in results for CBR and TCP flows is attributed to the reason that for TCP flows the medium control is more often taken away from the malicous node because of the transmission of the TCP-Ack packet by the AP. This reduces the amount of time for which the channel is occupied by a node and hence reduces the throughput.

Figure 9 (c) shows the attack impact of the relative location of the malicious nodes. In this case, where the malicious nodes (STA 4 and 8) are adjacent, the impact of the attack is stronger and thus results into higher overall throughput for the malicious nodes. The location of a node determines the channel conditions seen by it and hence the SNR. Spatial proximity leads to similar channel conditions and hence STA 4 and 8 have similar throughput variation while throughput variation is different when STA 4 and 5 are malicious.

### C. Beacon-Based PCF Attack

The beacon-based PCF attack is simulated under two different topologies, one consisting of 5 nodes and the other consisting of 8 nodes. In the normal fault-free case there is a single PC in the form of AP. The throughput shown in Figure 10 denoted as PC-5 or PC-8, represents the throughput in a non-adversarial environment.

In the malicious scenario we make a node to act as a PC along with the correct AP. The malicious node sends beacons to initiate the CFP, but restrains from sending the poll message. Figure 10 shows that the throughput in case of malicious behavior, denoted as Malicious-5 and Malicious-8, drops to nearly zero. During the entire simulation time of 200 seconds only 34 packets are generated and transmitted by the malicious node in the form of beacons, while over 2000 packets are lost by the correct nodes. Thus, the results indicate that this attack is low rate requiring very little complexity at the attacker and resulting in maximum damage to the correct nodes.

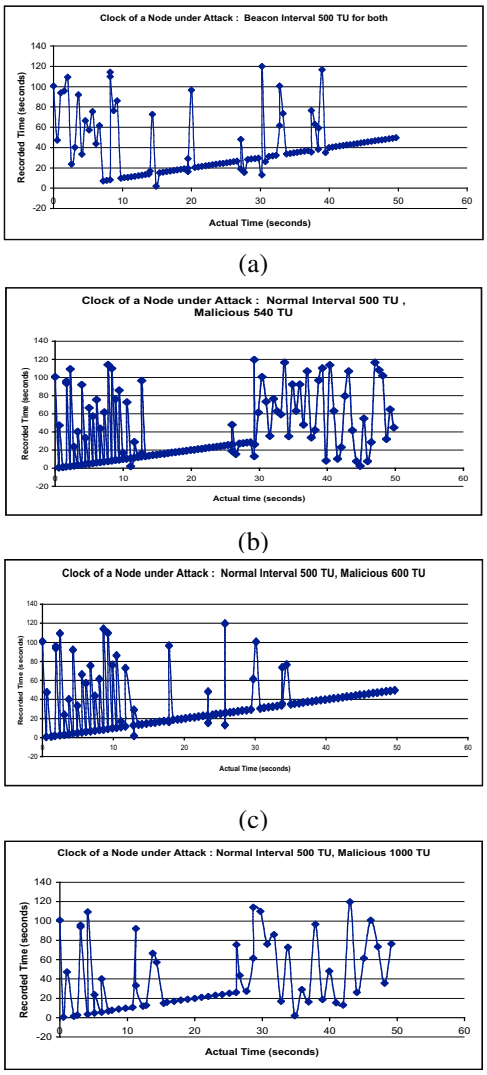



Fig. 11. Beacon-Based Synchronization Attacks

### D. Beacon-Based Synchronization Attack

We conducted simulations of the synchronization attack in a scenario where STA 1 runs a copy of a malicious MAC that sends beacon packets with incorrect time stamps only during the CFP period. Figure 11 plots the clock of normal nodes (under attack) versus the global clock. In case of normal operation the clock of the node must match the global clock and hence the plot must be a straight line without any jumps. In case of a malicious attack, the clock of a normal node can be deviated by sending a single beacon with a malicious time value. This error in clock will exist until a correct beacon is received by that node.

It is evident from Figure 11 that the larger the CFP period of a malicious node, the larger the fluctuations in the clock. This is because the malicious node sends more beacons with random time stamps during the large CFP period. In Figure 11 (d) the malicious node has a double CFP period compared to the correct PC causing the malicious node to send a lot more beacons. Thus it leads to more fluctuations in the clock. We note that this is a low rate attack: by sending only one beacon packet during each beacon interval, a malicious node can cause significant de-synchronization in the network.

The level of de-synchronization observed in Figure 11 may lead to failure of several features in other protocols which inherently depend upon synchronization. For example, the PSM mode depends upon synchronization between the AP and nodes which can be rendered useless by such a de-synchronization attack. De-synchronization can affect several protocols in wireless networks. Collective sleeping methodology [26] adopted in MAC protocols for sensor networks, would also fail under such an attack, which in turn can lead to high power usage and loss of communication amongst the neighbor nodes. Another example is the packet leashes [11] protection against wormholes attacks, which also depends on global synchronization amongst the nodes of the network. Thus, such an attack is not limited to 802.11a, b, and g, but can also be carried against several other wireless protocols.

## IV. Attack Detection Techniques

### A. NAV-Based Attack Detection

One method to detect the NAV attack, without using cryptographic mechanisms, is proposed in [13]. The proposal is to make nodes to listen to the next slot and verify if the transmission is still going on. The method does detect the attack, but has limitations such as preventing the implementation and deployment of any sleeping mechanism since the nodes have to always sense the channel. Another limitation is that it cannot detect an attacker who sends data throughout the alotted transmission time and keeps the channel occupied. Such a scenario can occur when a malicious node sends a large NAV and then transmits for only a few slots before ceasing its transmission. In this scenario the other nodes will have to listen to each slot before sensing the channel to be idle.

Our simulations demonstrate that by designing an appropriate protocol at higher levels the attack can easily be nullified. The experiments in Section III-A, show that if TCP is used at higher layers, then the AP tries to send the TCP acknowledgment back to the malicious node, causing the malicious node to relinquish the channel. If the malicious node gives up the channel, to regain it back, it has to again compete with all the nodes in the vicinity. A malicious node can only control the channel after it gets access to the channel once but using TCP causes it to give up the channel after a small time. This causes the malicious node to repeatedly try and contend for the channel thus annulling the attack. Our results indicate that MAC protocols must provide some form of flow control and fairness, potentially by coupling these mechanisms with transport layer protocols to guarantee an attack free operation.

### B. CSMA/CA-Based Attack Detection

A possible detection technique to this attack is proposed in [14]. The solution entails the receiver with the responsibility

of assigning the back off to the sender piggy backed on the CTS packet. The receiver can use a deterministic function or assign such a function to the sender to select the back off value. This approach may work in an infrastructure mode which has an in-built central point, however, it cannot be applied for an ad hoc network, in a distributed fashion. Consider a scenario where a node is receiving packets from multiple senders. In this case, the receiver node must design an optimal function which ensures fairness to all the senders, and then transmit it to the senders. This will consume precious bandwidth and power. In addition, this approach will not work correctly in a scenario where the receiver itself is malicious and sends large back off parameters to the sender throttling its bandwidth.

To address these concerns, we propose to use history information to determine if a node is malicious. Each node, when finding the channel busy, will record the identity of that sender and later use the information to calculate the probability of the channel being accessed for each node and try to determine whether it is random or not. This mechanism is completely distributed and can be run independently at each correct node. The history information can help in detecting malicious behavior in any node regardless of it being a sender or receiver. One advantage of this mechanism is that it is inexpensive since it relies on computation which is much cheaper to perform than transmissions in a wireless network. Because of the ease of use and low complexity, this method can be also used for ad-hoc and sensor networks which rely on CSMA/CA based mechanisms to avoid collisions. The latency of detection of such a mechanism may depend on how frequently the malicious node tries to access the channel. If the malicious node is using a smaller back off window but only transmitting intermittently, then the adversarial effect will be very small. In such a case state based approach might lead to longer latency in detection but the effect of the attack will also be unnoticeable. One can augment this approach with each node assigning a trust or reputation metric to every neighbor.

### C. Detection and Mitigation for Beacon-Based Attack

One way to prevent the attacks is to authenticate every beacon and to require a node to accept only authenticated beacons. This way only an authenticated PC or the AP can send beacons which can be used for time synchronization. Authentication can remove the attack vulnerability in the case when the attacker is an outsider, and is appropriate for centralized scenarios where only one node has the responsibility of sending the beacon. However, this method will not work correctly if the authentication protocol is compromised.

In a decentralized scenario (i.e. ad-hoc mode), where the time synchronization is performed in a distributed fashion, it is more costly to provide authentication for all the nodes when nodes are dynamically joining and leaving. One solution that can protect against the attack to some extent is to have each node maintaining a *guard time*. A node can only change its clock if the time stamp in the beacon and its clock difference is within the guard time assuming the clock drift is bounded. We implemented a modified MAC that maintains a guard of $\Delta$

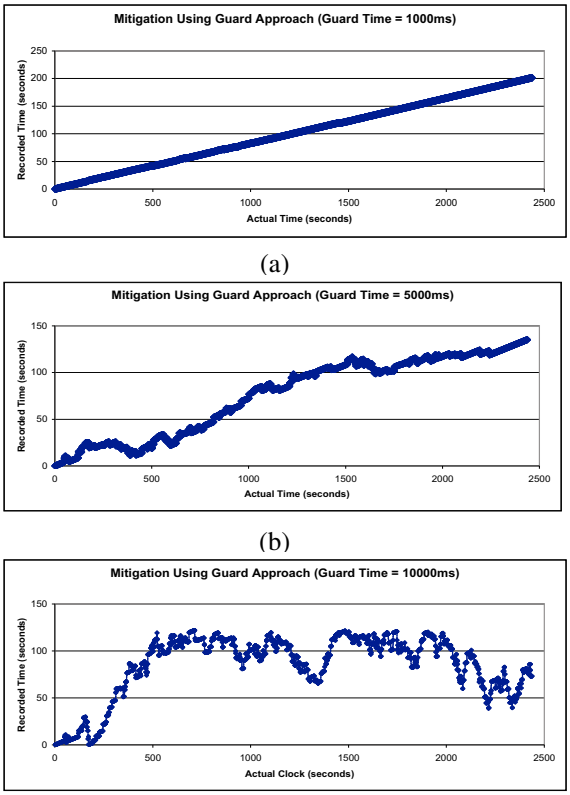
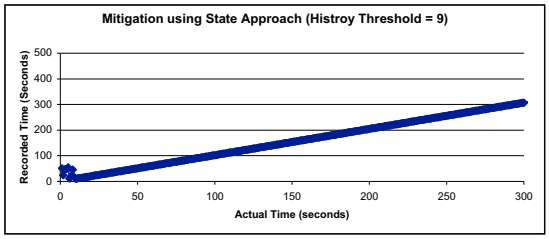



Fig. 12. Guard-Based Approach

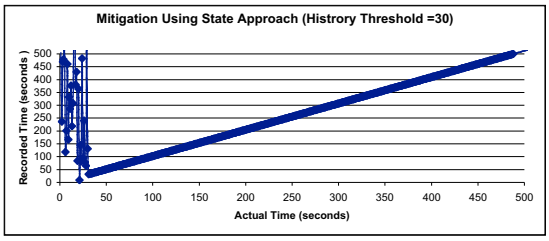
about the node's current clock. As it can be seen in Figure 12, the smaller the $\Delta$, the closer a node's clock matches the global clock. This mechanism has some limitations. If nodes join in later and clocks are not synchronized then this method can cause lack of synchronization between the new node and the existing network if the clocks differ more than the guard time. The guard time can decrease the effectiveness of the attack, but an attacker can still try to introduce a drift by repeatedly sending well crafted time stamps and slowly drifting the clock of the system i.e. cause *slow poisoning* in the system.

To address some of these limitations, we present another state based approach for prevention of synchronization attacks using beacons. In this approach each node maintains a state consisting of changes to time clock in the last time interval, say X. Along with X, a node maintains the identity of the node which sent that beacon. Each node maintains a history threshold $\rho$ which denotes the maximum number of times any node can cause changes to its clock. So if a node A sent more than $\rho$ beacons within X time which caused the clock of node B to change every beacon, then node B will mark node A as malicious. From that moment node B ignores any incoming beacon from node A. As in the above proposed CSMA/CA mitigation approach, this mechanism can run independently at each node and is completely distributed.

We implemented this mitigation in the 802.11 MAC pro-




(a)




(b)

Fig. 13. State-Based Detection Approach

tocol and performed the same synchronization attacks as in Figure 11 against the new modified MAC. One of the nodes in the network acts maliciously by sending beacons with random time stamps, while the correct nodes are using the 802.11 MAC with state based approach. The results depicted in Figure 13 show that initially a few malicious beacons are accepted and the local clock is changed to the time stamp in the beacon. Once the $\rho$ threshold is crossed, any further beacons from the malicious node are ignored.

One can also use the guard band approach in tandem with the state based approach to achieve much greater defense over the attack. For example, initial fluctuations which could be caused in the clock when a state-based approach is used, can be controlled by using a guard band as shown in Figure 12.

## V. Concluding Remarks

In this paper we demonstrate through simulations several channel access attacks against the 802.11b protocol. We analyze them by examining the effect of multiple attackers, their relative positioning and the influence of the choice of high level protocols. Our simulations show that when the transport protocol of choice is TCP, the effect of the attack is not so strong as when the transport protocol is UDP (for CBR flow). The relative location of the attackers also has an influence on the attack, when the protocol of choice is TCP.

In addition, we identify new synchronization attacks that affect the PSM and PCF components of 802.11. Our simulations indicate that these beacon-based attacks can create significant damage to a large number of nodes, without requiring a lot of work on the attacker side, i.e. they are low rate attacks. The level of de-synchronization observed in our simulations may lead to failure of several other protocols and services which rely on synchronization for correct operation.Synchronization attacks can affect a large class of wirless protocols and we are studying it further. Finally for every presented attack we propose and discuss several mitigation techniques. Our discussion indicates that cryptography-only mechanisms are not enough in protecting against such attacks and that trust and reputation based mechanism can help in mitigating the attacks. Issue of cross-layer design to achieve secure protocol is definitely a framework we are looking at.

## References


[1] IEEE, *IEEE Std 802.11, 1999 Edition*. 1999. http://standards.ieee.org/catalog/olis/lanman.html.

[2] IEEE, *IEEE Std 802.11b-1999*. 1999. http://standards.ieee.org/.

[3] S. Marti, T. Giuli, K. Lai, and M. Baker, "Mitigating routing misbehavior in mobile ad hoc networks," in *The $6^{th}$ ACM MOBICOM*, August 2000.

[4] Y.-C. Hu, A. Perrig, and D. B. Johnson, "Ariadne: A secure on-demand routing protocol for ad hoc networks," in *The $8^{th}$ ACM MOBICOM*, September 2002.

[5] Y.-C. Hu, D. B. Johnson, and A. Perrig, "SEAD: Secure efficient distance vector routing for mobile wireless ad hoc networks," in *The $4^{th}$ IEEE WMCSA*, June 2002.

[6] K. Sanzgiri, B. Dahill, B. N. Levine, C. Shields, and E. Belding-Royer, "A secure routing protocol for ad hoc networks," in *The $10^{th}$ IEEE ICNP*, November 2002.

[7] P. Papadimitratos and Z. Haas, "Secure routing for mobile ad hoc networks," in *SCS CNDS*, pp. 27–31, January 2002.

[8] B. Awerbuch, D. Holmer, C. Nita-Rotaru, and H. Rubens, "An on-demand secure routing protocol resilient to byzantine failures," in *ACM WiSe*, September 2002.

[9] P. Papadimitratos and Z. Haas, "Secure data transmission in mobile ad hoc networks," in $2^{nd}$ *ACM WiSe*, 2003.

[10] Y.-C. Hu, A. Perrig, and D. B. Johnson, "Rushing attacks and defense in wireless ad hoc network routing protocols," in *ACM WiSe*, 2003.

[11] Y.-C. Hu, A. Perrig, and D. B. Johnson, "Packet leashes: A defense against wormhole attacks in wireless ad hoc networks," in *INFOCOM 2003)*, April 2003.

[12] N. Borisov, I. Goldberg, and D. Wagner, "Intercepting mobile communications: The insecurity of 802.11," in *In $7^{th}$th MOBICOM*, July 2001.

[13] J. Bellardo and S. Savage, "802.11 denial-of-service attacks: Real vulnerabilities and practical solutions," in *In USENIX 2003*, 2003.

[14] P. Kyasanur and N. Vaidya, "Detection and handling of MAC layer misbehavior in wireless networks," in *DSN'03*, 2003.

[15] IEEE, *IEEE Std 802.11a-1999*. 1999. http://standards.ieee.org/.

[16] IEEE, *IEEE Std 802.11g-2003*. 2003. http://standards.ieee.org/.

[17] I. Akyildiz, M. Vuran, O. Akan, and W. Su, "Wireless sensor networks: A survey revisited," *To appear in Computer Networks Journal (Elsevier)*, May 2005.

[18] I. Akyildiz, X. Wang, and W. Wang, "Wireless mesh networks: A survey," *Computer Networks Journal (Elsevier)*, March 2005.

[19] A. Stubblefield, J. Ioannidis, and A. Rubin, "Using the fluhrer, mantin, and shamir attack to break wep," in *NDSS'02*, February 2002.

[20] IEEE, *IEEE Std 802.11i-2004*. 2004. http://standards.ieee.org/.

[21] C. He and J. C. Mitchell, "Security analysis and improvements for ieee 802.11i," in *NDSS'05*, February 2005.

[22] D. B. Faria and D. R. Cheriton, "Dos and authentication in wireless public access networks," in *ACM WiSe*, September 2002.

[23] M. L. Lough, *A Taxonomy of Computer Attacks with Applications to Wireless*. PhD thesis, Virginia PolyTechnic Institute, April 2001.

[24] V. Gupta, S. Krishnamurthy, and M. Faloutsos, "Denial of service attacks at the mac layer in wireless ad hoc networks," in *MILCOM'02*, October 2002.

[25] "The network simulator - ns2." http://www.isi.edu/nsnam/ns/.

[26] W. Ye, J. Heidemann, and D. Estrin, "An energy-efficient mac protocol for wireless sensor networks," in *INFOCOM 2002*, 2002.

[27] M. Raya, J. P. Hubaux, and I. Amad, "DOMINO: A System to Detect Greedy Behavior in IEEE 802.11 Hotspots," in *MobiSYS'04*, June 2004.