Available online at www.sciencedirect.com

ScienceDirect

DOI: 10.1016/j.comnet.2015.05.003 (2016) 1–21

Author's postprint

A novel cheater and jammer detection scheme for IEEE 802.11-based wireless LANs

Eduard Garcia-Villegas*, M. Shahwaiz Afaqui, Elena Lopez-Aguilera

Wireless Networks Group (WNG) - Department of Network Engineering, Universitat Politecnica de Catalunya (UPC) - i2CAT Foundation, Barcelona, Spain

Abstract

The proliferation of IEEE 802.11 networks has made them an easy and attractive target for malicious devices/adversaries which intend to misuse the available network. In this paper, we introduce a novel malicious entity detection method for IEEE 802.11 networks. We propose a new metric, the Beacon Access Time (BAT), which is employed in the detection process and inherits its characteristics from the fact that beacon frames are always given preference in IEEE 802.11 networks. An analytical model to define the aforementioned metric is presented and evaluated with experiments and simulations. Furthermore, we evaluate the adversary detection capabilities of our scheme by means of simulations and experiments over a real testbed. The simulation and experimental results indicate consistency and both are found to follow the trends indicated in the analytical model. Measurement results indicate that our scheme is able to correctly detect a malicious entity at a distance of, at least, 120m. Analytical, simulation and experimental results signify the validity of our scheme and highlight the fact that our scheme is both efficient and successful in detecting an adversary (either a jammer or a cheating device). As a proof of concept, we developed an application that when deployed at the IEEE802.11 Access Point, is able to effectively detect an adversary.

Keywords: IEEE 802.11, WLAN, jamming, cheater, security

1. Introduction

Wireless Local Area Networks (WLANs) occupy different sections of the 2.4 and 5GHz Industrial, Scientific and Medical (ISM) radio bands. ISM bands can be used freely at low transmission power without license, making them a very attractive alternative for building domestic wireless communications systems. This is both one of the keys for the success of Wi-Fi based WLANs, and the source of many interference issues affecting the operation of a WLAN. In recent years, growth in IEEE 802.11 WLAN technology has drastically increased due to its ease of deployment, convenience and cost efficiency. The IEEE 802.11 protocols were designed with the assumption that all the nodes that want to communicate, would follow specific predefined rule of engagement to transmit and receive data. These were not designed to withstand adversaries attacks intended to interrupt the transmission. The success of IEEE 802.11 has attracted more and more users to employ these networks, while increasing the potentials for attackers to operate.

With time, the wireless attacks on IEEE 802.11 have become more sophisticated and are evolving to counter every new development made in these networks. The most prominent of these attacks are layer-1 attacks which are seldom

[*]Corresponding Author: Eduard Garcia-Villegas; Email, eduardg@entel.upc.edu; Tel: +34 93 413 71 20; fax: +34 93 413 70 07

considered a threat because they are typically generated from non-Wi-Fi devices sharing the same ISM bands such as micro wave ovens, cordless phones, etc. These non-Wi-Fi devices, when located within a WLAN's coverage area, unintentionally radiate unwanted energy that can affect the whole network. Furthermore, most of the people are not familiar with the interference abilities of such devices and the people who are familiar do not have the control over their placement.

These attacks are further aggravated when done purposefully. An attacker/adversary with the intent to disrupt the network can use low-priced and readily accessible RF jammers. Such attacks can appear to be simple in nature but can have devastating consequences for corporate companies since those security breaches can break down the core communication line within a company (e.g. critical voice over Wi-Fi communication lines which require continuous Wi-Fi connection and email services) that can result in reduced productivity. The ease to attack IEEE 802.11 networks is indicated by [1] where the authors demonstrate the use of off-the shelf hardware that can be used to severely disrupt the network.

In [2], Xu et al. define an adversary as a *jammer to be an entity who is purposefully trying to interfere with the physical transmission and reception of wireless communications*. For the sake of simplicity and keeping in mind the adverse effects caused by non-Wi-Fi devices, we consider them also to be acting as jammers.

The jammer spreads energy over the targeted spectrum, where it becomes difficult to extract the desired signal from interfering signals. Furthermore, due to Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA) based channel access, the Wi-Fi networks become an easy target by these adversaries, where a jammer can even utilize low power to disrupt the network. IEEE 802.11 standard [3] provides different operating modes: Distributed Coordination Function (DCF), Point Coordination Function (PCF), Hybrid Coordination Function (HCF) with HCF Distributed (EDCA) and Controlled Channel Access (HCCA). The DCF is the mode currently employed in most deployments and uses CSMA/CA contention-based MAC algorithm. In this case, before initiating a transmission, a station senses the channel to determine whether it is busy during a period of time called the Distributed Inter-frame Space (DIFS). If the medium is sensed busy, the transmission is delayed until the channel is idle again, and a slotted binary exponential backoff interval is chosen in the range [0, CW-1], where CW is the contention window. The value of CW is set to its minimum value, CW_{min}, in the first transmission attempt and increases in integer powers of 2 at each retransmission, up to a pre-determined value CW_{max}. For each data frame successfully received, the receiver transmits an ACK frame after a Short Inter-frame Space (SIFS) period. The protocol described above is called the basic or two-way handshake mechanism. In addition, the specification also contains a four-way frame exchange protocol known as the RTS/CTS (Request to Send/Clear to Send) mechanism.

Due to CSMA/CA characteristics, this contention-based MAC mechanism is very sensitive to Denial of Service (DoS) attacks based on jamming techniques. This kind of attacks consists in the transmission of a powerful signal in the frequency band employed by IEEE 802.11 devices. Thus, the medium is always sensed busy during the jammer signal by IEEE 802.11 clients. Obviously, jammer influence will lead to very harmful effects in MAC protocol performance. Jamming attack in IEEE 802.11 can prevent the nodes to perform legitimate MAC operations or can cause the collision of frames that force repeated backoff which can even jam the complete transmission process. The jamming signal interferes and corrupts the desired signal in reception, while causing the co-channel transmitters to reschedule the transmission for longer period of time. Different factors are incorporated in the effectiveness of interference that a jammer creates namely distance between a jammer and a wireless device, transmission power of jammer and the network devices, and the MAC protocol used within the network.

Different attack strategies can be employed by a jammer while trying to interfere with other communicating nodes. In [2], the authors have differentiated jammers based on their attack model. They have defined four types of jammers namely constant jammers, deceptive jammers, random jammers and reactive jammers. According to the authors, a constant jammer continues to transmit radio signal without following any MAC layer protocol, a deceptive jammer continuously transmits regular frames without any gap, thus deceiving other communicating nodes to believe that a legitimate transmission is occurring, a random jammer that transmits for a time and goes to sleep, where both the transmission time and the sleep time can be random, and a reactive jammer that starts transmitting jamming signals as soon as it detects activity on the shared medium and goes to sleep when there is no one transmitting.

An intelligent jammer can also exploit the standard DCF that is used to coordinate nodes for medium access within IEEE 802.11. In [4] Pelechrinis et.al. define intelligent jamming models and methods used to jam IEEE 802.11 networks. In [5], the authors investigate the fabricated CTS attack to the MAC scheme of IEEE 802.11 and propose a mechanism to prevent such attack. This attack is based on a jammer acquiring the use of shared channel by transmitting a fabricated CTS signal, which contains large Network Allocation Vector (NAV) to falsely defer transmissions from other users for longer duration.

A jammer can also be a cheating device that misuses the IEEE 802.11 MAC constraints in order to attain bandwidth gains. This device can have the ability to choose Clear Channel Assessment (CCA) threshold, backoff window size and/or inter-frame space. By increasing the CCA threshold, the cheating device can improve its opportunity to transmit and thus can effectively disable channel sensing. It can continue to transmit over the medium, while causing other transmitting stations (STA) to undergo collisions and thus backoff from transmitting. The cheating device can also observe collision but the backoff period is kept shorter (is not frozen because carrier sensing is already disabled). The authors in [6] extensively explain how a selfish station with higher CCA can experience bandwidth gains. Similar bandwidth advantages can also be achieved by utilizing a smaller contention window, which helps the cheating node to backoff for smaller periods than average, when collisions occur. The cheating device can also maneuver to cheat the IEEE 802.11 MAC constraint by reducing its DIFS. By reducing the DIFS, the cheating station can gain quick access to the medium, thus depriving other stations from their fair share.

Therefore, finding solutions to eliminate jamming is very important in IEEE 802.11 networks. This solution can only be found by first enabling the network to detect the jammer and then to find an appropriate solution to counter such threats. In this paper, we introduce a novel malicious entity detection method for IEEE 802.11based WLANs. This method exploits the fact that beacon frames are prioritized over other transmissions and thus the corresponding Beacon Access Time (BAT) lies within definite bounds. Hence, BAT can be used to detect ambiguous conditions within the wireless network. The method to detect a malicious entity relies mainly on the observed change in BAT. Most of the previous detection schemes, summarized in section 3, take into account the transmission Packet Delivery Ratio (PDR), Carrier Sensing Time (CST) and Received Signal Strength (RSS). To the best of our knowledge, no previous work has been done to detect a malicious entity in IEEE 802.11 WLANs based on beacon frame analysis. Furthermore, our proposed technique is also the first mechanism that has proved to be capable of detecting jammers as well as cheating devices within Wi-Fi networks. In more detail our contributions in this paper are as follows:

- We investigate the impact of different jammers in IEEE 802.11 based WLAN networks by utilizing both simulations and real Wi-Fi devices. While previous studies have considered the impact of malicious devices on WLAN network, we provide a clear insight about the problem in hand.
- We define a new metric called Beacon Access Time (BAT), and we provide an analytical model for it that is evaluated through simulations and real measurements. The proposed analytical model is used to validate the utility of the BAT metric.
- We design and implement a novel jammer and cheater detection method based on BAT measurements and predictions.
- We perform simulations and measurements to investigate the performance of our detection technique. By extensive experimentation, we prove that: one, BAT can be predicted by an access point (AP) under normal condition; and two, BAT values become considerably different in the presence of a jammer and thus the AP can sense the presence of a malicious entity and take necessary actions.
- With the help of simulations, we evaluate BAT detection mechanism in the presence of a selfish node called cheater that wants to quickly acquire access of the shared channel. The results indicate that the cheater can be sensed by the AP using BAT.
- We implement a BAT based jammer detector in a Linux based IEEE 802.11g AP as a proof of concept. The results verify that the BAT predictions made by the developed application follow the analytical BAT model presented in this paper and are useful to detect the presence of a jammer.

The rest of the paper is organized as follows. We briefly review the impact of different jammers on an IEEE 802.11 network in section 2. In section 3, we introduce our BAT based malicious entity detection scheme. After a related work review, this section provides an analytical model to predict BAT, simulation and experimentation setup, and the analysis that validates our scheme. In section 4, we describe the jammer and cheater detection abilities of our scheme with the help of simulation and real measurement results. Finally in section 5, we present our conclusions and future line of research work.

2. Jamming an IEEE 802.11 WLAN

As anticipated in the previous section, the effects of a jammer on a WLAN depend on several factors, such as the strength with which the jammer signal is received, the modulation and coding scheme used by the WLAN STAs, the frame size and the role of the attacked device (transmitter or receiver). In this section, we perform several experiments to study these factors when an IEEE 802.11 link is interfered by a channel-oblivious, memoryless, continuous jamming device.

2.1. Effects of a jammer

Due to the CSMA-like MAC used in IEEE 802.11 WLANs, the effects of interference on a receiver are very different from the effects of interference on a transmitter. In the first case, the jamming signal is added to the desired signal thus reducing the effective Signal to Interference plus Noise Ratio (SINR) and, therefore, increasing reception errors. The use of robust modulations can mitigate the impact of this kind of interference. On the other hand, when the device under attack is a transmitter, the detection of any signal above the energy detection (ED) threshold on the current channel will defer the desired transmission. In order to study the effects of the jammer on an IEEE 802.11 transmitter and on a receiver, we separate the STAs reception and transmission paths with the RF circuit depicted in Figure 1

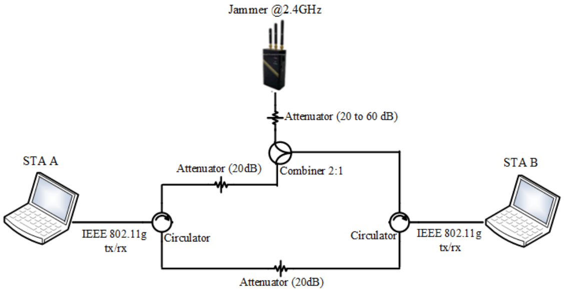


Figure 1. Testbed setting used to study the effects of a jammer on an IEEE 802.11 link.

In the first set of experiments we used a signal generator[1] as a jamming device. After exhaustive measurements with different configurations, the swept sine function[2] was found to be the most effective jamming signal (produces the most harmful effects using the least amount of energy). A deeper discussion on this fact is omitted due to space constraints. The STAs either transmit or receive UDP datagrams[3] at the maximum allowed rate using the *iperf*[4] tool. Application layer throughput was measured at the receiver. Only STA B receives the jamming signal; when B is acting as a receiver, A's data frames are mixed with the jamming signal; when acting as a transmitter, the jamming signal is mixed with A's ACK frames.

We summarize our observations in Figure 2. First of all, the effects of the jammer start being observable when the power of the jamming signal is close to the receiver sensitivity (-87dBm for 6Mbps); then, as the power of the jamming signal is increased, it gradually degrades the performance of the WLAN link to the point at which frames cannot be successfully decoded. This point requires more energy when a robust modulation is used (cf. 6Mbps vs. 54Mbps lines in Figure 2). After different experiments we can also conclude that the frame size used by the STAs does not show a definite influence on the effects of the jammer.

Previous experiments with IEEE 802.11b equipment [7] concluded that, for robust modulations, it is more effective to target the transmitter's CCA; on the contrary, a faster modulation is more sensitive to packet errors produced at the receiver. However, according to our observations in the testbed depicted in Figure 1, attacking an OFDM-based IEEE 802.11 transmitter is clearly more effective than attacking a receiver regardless of whether a robust or a fast

[1] Agilent ESG-D

[2] A sine waveform whose frequency varies to sweep the whole 2.4GHz band

[3] TCP is not being used in order to avoid the influence of TCP congestion control and retransmission mechanisms in the jammer evaluation.

[4] http://iperf.sourceforge.net/

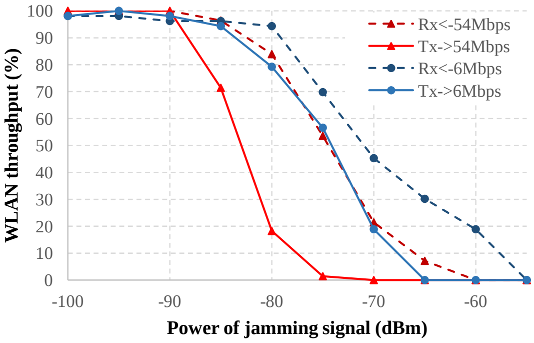


Figure 2. Throughput of an IEEE 802.11 link when the receiver/transmitter is under jamming.

modulation is used. Note that the jammer affects the link in two ways: from the transmitter perspective, the jammer occupies the channel thus preventing data transmissions (exposed node problem), and second, it prevents successful reception of the receiver's ACK frames. Therefore, attacking the transmitter is effective even when the received jamming signal is below the ED threshold. On the other hand, the receiver under the jammer attack is affected by the hidden node problem (i.e. transmitter assumes that the channel is idle and transmits while it is actually busy within receivers vicinity) and receives corrupted frames; it is also affected by the exposed node problem when attempting to send ACK frames back to the transmitter while the medium is occupied by the jammer. However, an ACK frame is more likely to be sent than a data frame in the presence of the exposed node problem since it requires the channel to be clear during a shorter period (SIFS < DIFS).

2.2. *Recovery after a jammer attack*

After observing the harmful effects of the jammer, even at relatively low transmission power, in this section we study what happens afterwards, that is, when the jammer is deactivated. In order to do so, we run several experiments in which we switch on a jammer only for a given time and then observe how the WLAN recovers its operation. More precisely, we evaluate the time required by the STAs and AP to re-associate after a jamming attack by varying different set of hardware and software combinations. Figure 3 elaborates the setup used to perform the experiments.

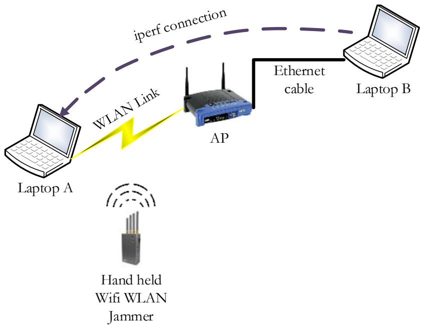


Figure 3. Experimental setup to understand the impact of a jammer in IEEE 802.11 network

A simple Wi-Fi network was established in an isolated environment. Both ends of the radio link (AP - STA) were subjected to the interference created by the jammer. Two different APs equipped with different radio chipsets (Linksys WRT54G based on Atheros AR9103 and TP-Link WR1043ND using Broadcom BCM2050) were used in order to analyze the impact of the jammer on two different hardwares. Both the APs were running in IEEE 802.11g mode and operated on channel 1 (i.e. 2.412GHz). Two laptops (laptop A and laptop B) were used in the experiments that had built-in IEEE 802.11a/b/g NICs. Laptop B was connected to the AP through Ethernet cable and, additionally, it was equipped with PCMCIA based IEEE 802.11a/b/g NIC that was used to sniff wireless traffic. Laptop A was connected to the AP through IEEE 802.11g based wireless connection. A UDP stream was established with the help of *iperf* network testing tool between laptop A and laptop B. UDP datagrams were generated from the client (i.e. laptop B) and were sent to the server (i.e. laptop A).

The jammer used in these experiments was a portable handheld broadband jamming device named CVSAL3405, which is capable of interfering in the following bands: 895-1000MHz, 1195-1300MHz and 2395-2500MHz. The total output power on its three omni-directional antennas was 450mW, enough to prevent any communication within a radius of 20m over the specified bands. It is fitted with an On-Off button to switch on and off the interference. The jammer had the characteristics of ignoring the IEEE 802.11 MAC procedures and could constantly transmit energy on the channel when switched on. In the experiments, the jammer was switched on (for a particular time) while frames were being sent from laptop B to laptop A. Once the jammer was activated, the wireless link was completely broken. When it was switched off again, laptop A and the AP tried to recover their link. The sniffer was used to capture the frame stream and to analyze the sequence of events that followed a jamming attack.

In order to make the experiments more observant, two different operating systems (i.e. Microsoft Windows 7 and Linux Ubuntu 12.0.4) were used at laptop A. Additionally, the experiments were also repeated for the cases where the laptop A was using the built-in NIC and in other experiments it was using an external USB Wi-Fi device (i.e. TP-LINK TL-WN822N) instead. Linux operating system was also used in laptop B for all the experiments. Table 1 portrays the combination of AP, operating system and NIC (used by laptop A) within our Wi-Fi network.

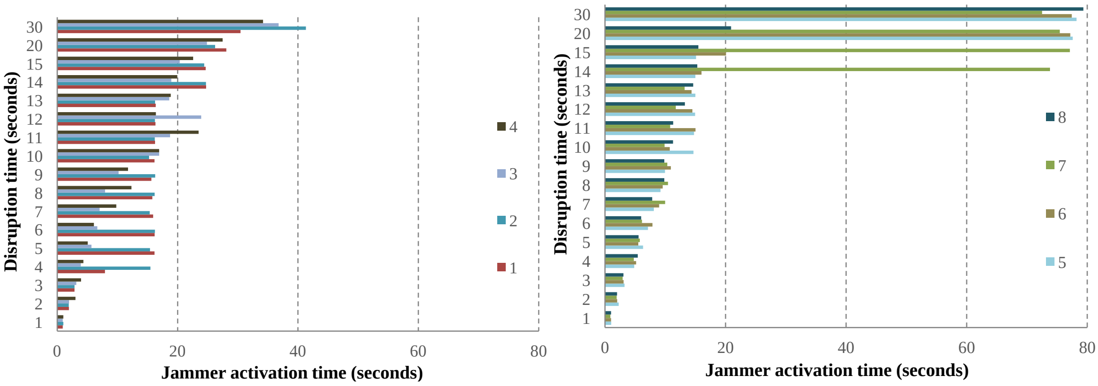
Table 1. Combination of hardware and software used to perform the experiments.

Combination	**Operating System used at laptop A**	**NIC used by Laptop A**	**AP used in Wifi Network**
1	Linux	Built-in	Linksys
2	Linux	Built-in	TP-Link
3	Linux	USB	Linksys
4	Linux	USB	TP-Link
5	Windows	Built-in	Linksys
6	Windows	Built-in	TP-Link
7	Windows	USB	Linksys
8	Windows	USB	TP-Link

For a particular experiment, the jammer attack lasted for a fixed specific time. For each combination of hardware and software, different experiments were performed where each one had different jammer activation time. The trace of frames captured by the sniffer before, during and after the attack depicted the impact of the jammer on the Wi-Fi link and was used to find the disruption time caused by the jammer. This disruption time was calculated by finding the difference between the last acknowledged UDP datagram sent before the jammer was activated and the first acknowledged datagram after the jammer was deactivated. In few of the experiments, the UDP data stream was dropped. I those cases, the instant at which association or re-association succeeded was considered to calculate the disruption time. Figure 4 shows the results of the experiments when eight different combinations of software and hardware (see Table 1) were used.

As the jammer activation time was increased, the disruption time also increased. It is interesting to note that when the attack lasted for 10 or more seconds, the disruption time increased more rapidly than the jammer activation time. Additionally, the disruption time for the case when the laptop A utilized Linux operating system was less than the case where the laptop A used Windows operating system. Furthermore, with jammer activation time higher than 14 seconds, the network manager of Windows operating system assumed the interface to be down and waited for some time before sending the association request to the AP, while Ubuntu's network manager immediately tried to re-establish the link even after long disruption times. This finding is more evident in Figure 4 when the jammer was active for 30s: the reassociation strategy employed by Ubuntu's network manager make it more resilient towards jammer activity.

Finally, No differences were observed when the AP device was changed. This leads to the conclusion that, since association and re-association are client-driven mechanisms, a simple strategy implemented in the client's wireless driver or network manager software can help communications to recover early after a jammer attack.



(a) Laptop A utilizing Linux operating system.

(b) Laptop A utilizing Windows operating system.

Figure 4. Disruption time in connection of laptop A to AP caused by the jammer.

3. Jammer and cheater detection in IEEE 802.11 WLANs

Jamming detection relies mainly on observed characteristics and relates them to each other to make a decision. Xu et al. [2] use PDR and Packet Sent Ratio (PSR) metrics. Different jamming attack models that may require different detection strategies are also defined in [2]. PSR corresponds to the number of frames transmitted by the sender divided by the actual number of frames that the transmitter wanted to transmit. The intended and the actual number of frames sent are different in the presence of a jammer. The jamming signal occupies the medium for long periods deferring legitimate transmissions and thus causes the buffer at the MAC of the transmitters to overflow (causes new frames to be discarded and old frames to be timed out). At the receiver side, PDR is computed. It is defined as the ratio of successfully received frames at the receiver to the total number of frames transmitted by the sender. If a jammer is present near to the receiver, frames might not be decodable at the receiver, and thus PDR value is degraded.

In [8], the authors utilized the propagation characteristics of the wireless channel to expose the presence of jamming devices. The authors use PDR to estimate the presence of a jammer. Whereas the authors in [9] proposed a least-squares (LSQ) based localization algorithm that estimates the jammers location.

A jamming activity can also be detected within the network by auditing current measurements. In [10], the authors use anomaly-based intrusion detection algorithm to investigate the presence of a jammer. The intrusion detection algorithm uses SINR based statistical analysis. The trace of SINR is collected from the receiver as well as two monitors (specially placed in the network to collect trace). The authors have proposed the execution of simple threshold based algorithm and a cumulative sum (cumsum) type algorithm locally on the monitors to detect changes in statistical characteristics of SINR. These algorithms were evaluated based on the detection probability, false alarm rate, average detection delay and their robustness to different detection threshold values. Furthermore the authors utilize the Dempster−Shafer (DS) algorithm to perform collaborative detection where the outputs of local detection algorithms are fused. It was found by the authors that the use of DS algorithm considerably increases the performance of local simple algorithm. The jammer model used by them was an off-the-shelf periodic on-off jammer that transmits only in the neighboring channel to the channel being used by the legitimate nodes. Furthermore, implementation of such scheme can be costly in terms of number of specific monitors required as well as the study on their placement within the network.

The malicious activity of a jammer can also be detected by measuring RSS along with the calculated PDR. In [11], the authors propose a scheme to detect a jammer based on PDR along with RSS and a mechanism to reduce the impact of jammer in IEEE 802.11 networks. The authors have justified the presence of a jammer by utilizing consistency check when RSS is high and the PDR is low. This scheme uses the jammed channel by adapting the modulation and coding scheme of each node based on successful transmission probabilities. The authors prove that the rate adaptation algorithm (RAA) improves the PDR and link utilization in presence of a jammer. Similarly, recent works have also investigated the impact of jamming strategies on IEEE 802.11 RAA; in [12], the authors characterize the effect of power control and rate adaptation to mitigate the effect of jammers; in [13], the authors have investigated the vulnerability of different RAA against jamming attacks and propose new methods to mitigate them. In [1], authors have utilized cumulative-sum algorithm to detect abrupt changes in SINR. They show that the output of their proposed algorithm increases in the presence of a jammer. Moreover, they propose to use the ratio of corrupted packets over correctly decoded packets as the cumulative-sum metrics to detect MAC layer attacks along with the SINR based cumulative-sum algorithm to detect physical layer attacks.

However, detection mechanisms based on RSS or SINR and PDR, can incur in false detections. First, according to [14], the assumption that a high RSS or SINR are synonym of a high PDR does not always hold. Second, mobility within the communicating stations formulates time varying distribution of received signal strength which makes it difficult to set reliable RSS-based thresholds. Moreover, it has been shown by authors in [15] that the jammer can affect the normal operations of a CSMA/CA based wireless network, by utilizing low power jamming signals which will not affect RSS measurements significantly.

In [2], the authors also proposed the measure of the CST (the time a station waits for the channel to become idle) when a jammer is present near to the transmitter. But the CST value does not only increase in the presence of a jammer, it can also rise with high number of transmitters. The authors in [7] define a jammer detection method based on the inspection of the number of transmission attempts per frame and use it to correspond to CST. The authors define a ratio T_f that is based on the total number of frames that have been sent to the channel and the total number of transmissions that were deferred to avoid collision. The access point measures load (occupation time) and T_f periodically; if T_f is above the value expected for that load, a jammer is present. The authors propose to utilize cell breathing method to reduce the effect of the jammer on an AP. If the load on an AP is high, the cell size of that AP is reduced (so as to allow a minimum number of nodes to connect to that particular AP), while the nodes may be able to connect to other APs that do not have a jammer present in its vicinity.

In [6], [16] and [17], the authors propose cheater detection methods in WLANs. The mode of detection in [6] and [16] is based on extensive monitoring and analysis of shared frames by the AP with the help of additional modules. In [17], the authors have designed a lightweight fair-share cheater detector mechanism that does not rely on the idle time distribution, but the proposed mechanism is only theoretically analyzed and the authors have not discussed the actual implementation aspects of their scheme.

Table 2 summarizes the main characteristics of the approaches found in the literature and offers a sneak peek of our proposal, the BAT-based detector. It provides the functionality comparison of the proposed system (which is explained in detail in the following sections) with some of the notable existing malicious entity detection systems. The comparison was done based on the detection capabilities, principle of detection, implementation requirements and approach followed by each detection method. It is pertinent to highlight that no single technique was found to detect both the jammer and the cheater together.

Note that additional hardware is required in [10] to measure SINR within the network. In [7], the authors propose to use CCA channel busy indication for detection process. But this method has an apparent drawback that the channel busy time can increase in the presence of a jammer as well as with increased number of active users. Detection methods in [2], [8] and [9] require extra signaling, which entails and increased overhead. The cheater detection schemes [6] and [16] require extensive monitoring that can lead to increased complexity at the AP. The cheater detection scheme proposed in [17] is based on theoretical analysis and thus cannot be compared with our proposed scheme. In comparison to these schemes, our proposed BAT-based scheme is a novel idea of detection and only requires the need to monitor beacons transmission at the AP; it is a cell-centric scheme and does not require additional signaling nor imposes hardware constraints.

3.1. Beacon Access Time

One of the most effective jammer detection mechanisms described in section 3 was based on CST measurements [2]. However, CST is highly dependent on the load of the cell due to the CSMA scheme defined by the IEEE 802.11; for example, the presence of a large number of legitimate transmitters will increase the measured CST given that a STA's transmissions may be deferred by an uncertain number of preceding frames that underwent a shorter backoff, had a higher priority, etc. As discussed in the following, if we measure CST but only for Beacons, we can keep those measurements between definite bounds, regardless of the cell load, due to the fact that beacons are prioritized over other transmissions.

Beacon frames serve a variety of functions, the most obvious of which are to identify an AP and to describe its capabilities. Notwithstanding, one of beacons' most relevant functions corresponds to their contribution to the Timing

Table 2. Functionality Comparison

	Jammer detection	**Cheater detection**	**Principle of detection**	**Implementation requirements**	**Approach**
BAT-based detector	Yes	Yes	Difference between TBTT and actual beacon transmission time	Additional software required at AP	Centralized
Xu et al. (2005) [2]	Yes	No	RSS and PDR based consistency check calculations	Extra signaling required for sending heart beat beacons	Distributive
Garcia-Villegas et al. (2010) [7]	Yes	No	CCA channel busy indication, TxDeferred transmission and load	Additional software required at AP	Centralized
Pelechrinis et al. (2009) [8]	Yes	No	PDR calculations	Extra signaling required for generation of probe signal to calculate PDR	Distributive
Liu et al. (2012) [9]	Yes	No	PDR calculations	Extra signaling required for generation of probe signal to calculate PDR	Centralized
Fragkiadakis et al. (2013) [10]	Yes	No	SINR calculations	Additional hardware required (monitors) for SINR calculations	Local and distributive collaborative
Ju and Chung (2012) [11]	Yes	No	RSS and PDR based consistency check calculations	Extra signaling required for generation of probe signal to calculate PDR	Distributive
Pelechrinis et al. (2009) [6]	No	Yes	Throughput monitoring module and low power probing module	Additional software required at AP	Centralized
Raya et al. (2004) [16]	No	Yes	Traffic traces collected to analyze scrambled frame or manipulated protocol parameters	Additional software required at AP	Centralized
Tang et al. (2014) [17]	No	Yes	Collision estimation and fair share detector for real-time backoff misbehavior detection	Theoretical analysis, no practical implementation	Centralized

Synchronization Function (TSF) [3]. STAs in the same Basic Service Set (BSS) are synchronized to a common clock. In an infrastructure BSS, the AP becomes the timing master for the TSF by periodically transmitting beacon frames that contain the AP's timestamp in order to synchronize the timers of other STAs in that BSS. Due to this function, beacons must be prioritized over other frames.

Beacons are sent according to the Beacon Interval parameter (typically around 100ms), defining a series of target beacon transmission times (TBTT). At each TBTT, the AP schedules a beacon frame as the next frame for transmission, i.e., the beacon is pushed to the first position of the AP's transmission queue, overtaking any other pending frame. The transmission of a Beacon complies with the IEEE 802.11 standard access, and hence it might be delayed due to CSMA deferrals. However, beacon transmission queue's CW is kept to 0, which effectively disables the backoff procedure, and uses PIFS instead of DIFS. This gives beacons priority over any other transmission in the BSS and, in consequence, if at TBTT the medium is busy, the beacon is the first frame to be transmitted in the BSS after the channel is released.

Table 3. Constants for IEEE 802.11g/n and a/n.

		IEEE 802.11g/n	11a/n
σ	Slot time	9μs	9μs
SIFS	Short Interframe Space	10μs	16μs
PIFS	Point Coordination Function InterFrame Space	SIFS+σ=19μs	25μs
DIFS	Distributed Coordination Function Interframe Space	SIFS+2× σ=28μs	34μs

For the above reasons, we define Beacon Access Time (BAT) to implement our malicious entity (cheater or jammer) detection scheme. At the AP, BAT is measured from the time at which the beacon is generated and placed at the head of the transmission queue (i.e. at TBTT), until the actual frame transmission start time. Figure 5 depicts the relationship between BAT, TBTT and Beacon Interval.

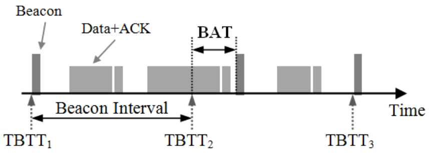


Figure 5. Representation of Beacon Acess Time (BAT).

3.2. *An analytical model for predicting BAT*

After presenting the basic concept of BAT-based malicious entity detection scheme, in this section an analytical model is proposed. This model is required to validate our hypothesis that BAT is indeed a good jammer detecting scheme. Furthermore, this model provides BAT estimations that are used as a reference by the detector to compare with the variations in actual measured BAT.

Note that, when traffic is all downlink, BAT will just remain constant at PIFS and, therefore, our interest is focused on studying uplink traffic conditions. In a very simple scenario, where ***N*** client STAs use the same PHY rate ***r*** [Mbps] and continuously transmit (i.e. in saturation) fixed-length frames of duration $T_{message}$, the BAT samples can be considered as a random variable that will depend on the channel status at TBTT. If N is sufficiently large, we assume that two consecutive data frame transmissions are spaced by one DIFS since the probability that all STAs have their backoff counter greater than 0 at the same time is small. Also, recall that any STA must sense the channel idle during DIFS seconds before attempting to transmit whereas the AP waits for a shorter time, PIFS, before sending a beacon frame. The difference between DIFS and PIFS is one empty slot time, σ (c.f. Table 3). Therefore, when TBTT coincides with the end of a previous transmission or comes earlier than σ after the last transmission, BAT equals PIFS; i.e. the beacon will overtake any other pending transmission. When TBTT is exactly σ seconds after the previous transmission, and there is a client STA with its backoff counter set to 0, that STA's frame will collide with the beacon. For any TBTT arriving later than σ seconds after the last transmission, the corresponding beacon will be deferred because a new data frame will occupy the channel before the AP senses the medium idle during PIFS. The BAT values will follow the behavior depicted in Figure 6. Assuming that the next TBTT can be any moment between consecutive transmissions (spaced by DIFS) or during a frame transmission, the average BAT ($\overline{BAT}$) can be deduced from Figure 6 as follows in (1).

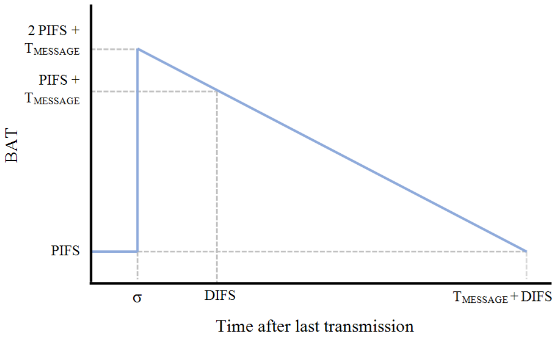


Figure 6. BAT values depending on the difference between last frame transmission end and next TBTT.

$$\begin{aligned}\overline{BAT} &= \frac{PIFS(T_{message}+DIFS)+0.5\times(T_{message}+PIFS)^2}{T_{message}+DIFS}\\ &= PIFS+\frac{1}{2}\frac{\left(T_{message}+PIFS\right)^2}{T_{message}+DIFS}\end{aligned} \tag{1}$$

$$T_{message} = T_{data} + \delta + SIFS + \delta + T_{ACK} \tag{2}$$

where SIFS, DIFS and PIFS are given in Table 3. δ represents the propagation delay, which can be neglected at typical WLAN distances. T$_{ACK}$ is the duration of an ACK frame and T$_{data}$ is the transmission time of a frame which depends mainly on the size of the payload and the PHY rate. For IEEE 802.11g transmissions, T$_{data}$ is:

$$T_{data} = T_{preamble} + 4\left\lceil \frac{(22 + (L_{header} + L_{data}) \times 8)}{4r} \right\rceil \\ + T_{SignalExtension} \tag{3}$$

where T$_{preamble}$ is 20μs, T$_{SignalExtension}$ 6μs and L_{header} is 36 Bytes. Note that T$_{message}$, as defined in (2), corresponds to legacy IEEE 802.11 transmissions; T$_{message}$ can be easily adapted to account for multiple frames within an EDCA TXOP or in the case of aggregation. Still assuming saturation conditions but in a more general scenario, where STAs may use different rates and different frame lengths, we use $\overline{T}_{message}$, the expected duration of a frame when the channel is found busy, which depends on the distribution of the different STAs' frame durations (T$_{message_i}$).

$$\overline{T}_{message} = \sum_{i=1}^{N} P_i T_{message_i} \tag{4}$$

where P$_i$ is the probability that the channel is occupied by STA$_i$ at TBTT. Using T$_{message}$ from (2), we can compute P$_i$ as follows:

$$P_i = \frac{T_{message_i}}{\sum_{j=1}^{N} DIFS + T_{message_j}} \tag{5}$$

Similar to [18], in (5) we assume that the IEEE 802.11 MAC maintains fairness in terms of access probability independently of the rate and bandwidth requirements of each station. When compared against simulation results (cf. section 3.3.2), this approach provides a good estimation of the expected BAT (relative error < 1.3%) when ***N*** is sufficiently large (above 10 STAs) and all STAs are in saturation. In those scenarios, the probability that all STAs are in backoff at the same time is very low and, therefore, the influence of the backoff procedure on the BAT is negligible. However, when the number of stations is smaller, the probability that the medium is idle for more than DIFS is higher due to the simultaneous backoff of multiple STAs and it does have an impact on BAT. Also, in the event of a collision, T$_{message}$ is reduced due to the absence of an ACK frame. In order to account for the effect of the backoff we also need to consider collisions and frame losses. As a result, we use P$_{busy}$, defined as the portion of time the channel is busy, which can be used likewise to model unsaturated conditions. In saturation, P$_{busy}$ can be written as in (7):

$$\overline{BAT} = PIFS + \frac{1}{2} P_{busy} \frac{\left(\overline{T}_{message} + PIFS\right)^2}{\overline{T}_{message} + DIFS} \tag{6}$$

$$P_{busy} = 1 - \frac{(1 - P_{tr})\,\sigma}{E_s} \tag{7}$$

P$_{tr}$ is the probability that at least one station will transmit during a randomly chosen slot time and E$_s$ is the average duration of a slot time, which includes empty slots, successful and unsuccessful frame transmissions. Please, refer to [19] for more details on those parameters. Contributions of [20] can also be used to extend this BAT prediction model to include the effect of transmission errors. As proven in 3.3, our assumptions do not lead to significant estimation errors.

3.3. Evaluation of BAT in different scenarios

In this section we study the different parameters that affect BAT under normal operation of the AP and its associated STAs, that is, when no jammer or cheater is present. In this case, BAT values will depend on the physical transmission factors associated with the active stations: number of stations, size of transmitted frames, physical transmission rate and offered load. This study aims to prove that BAT is predictable and that the model detailed in section 3.2 is accurate in predicting BAT.

3.3.1. Simulation environment

We employ a custom-made event-based simulation software tool implemented at the Universitat Politecnica de Catalunya (UPC). Our simulation program has been written in C++ programming language and follows all of the IEEE 802.11 MAC protocol details, emulating as closely as possible the real operation of each element (user stations and access points). It allows the evaluation of different parameters, such as throughput and BAT values, in heterogeneous scenarios. Moreover, it includes the emulation of non-legacy stations and a jammer element. The correct operation of the simulation tool was verified by comparing the results obtained with the information published in [19], under identical simulation conditions. It has also been employed in published papers [21][20].

3.3.2. Simulation and analytical results

The scenarios considered for this evaluation consist of a single AP serving a varying number of stations operating at different bit rates and transmitting frames with different payload sizes. All the figures included in this section present analytical and simulation results.

Figure 7 presents the complete trend of BAT values when different number of stations are used employing different rates and payload sizes within a cell under saturation conditions. In this case, STAs employing higher transmission rates or shorter payload size result in smaller BAT value. It would take less amount of time to finish the transmission, and thus the AP would have to wait for shorter periods before sending its beacon frame in the event that the medium is sensed busy at a given TBTT.

BAT value is also dependent on the number of stations actively competing for the medium with the AP within a cell. The effect on the BAT value is more observant when less than 10 stations are communicating to an AP. On the other hand, BAT performance tends to converge as the number of transmitting stations increase. This is due to the fact that the probability that the channel would be sensed busy at corresponding TBTT is closer to 1 when more than 10 stations are active in a cell. This dependency on the number of stations is more evident when large frames are used, whereas the BAT value shows a faster convergence when stations send shorter frames. If an AP finds the medium frequently occupied by large frames, the average BAT value converges slower due to the increased variance in measured BAT samples. On the contrary, when small frames or no frames at all are found at TBTT, BAT measurements are less variable and thus convergence is faster. From Figure 7 we also observe that the analytical model proposed in section 3.2 is accurate: analytical results coincide with simulation results with an error below 2%.

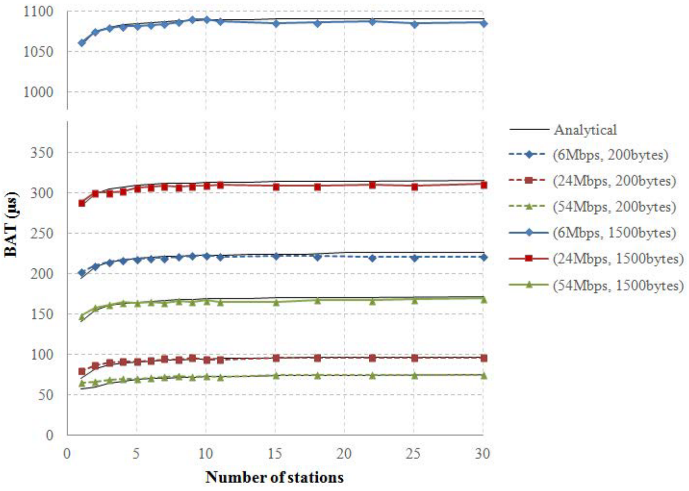


Figure 7. BAT values for different number of stations, transmission rates (6, 24 and 54Mbps) and payload sizes (200 and 1500Bytes).

BAT performance also depends on the cell load. Figure 8 corresponds to the scenario where 8 stations are communicating to an AP with a constant physical transmission rate of 18 Mbps and a constant payload size of 1500 Bytes. In this case, the traffic offered by the stations is increased gradually to the saturation point. Saturation load conditions begin at the value of 12.6 Mbps. The BAT values before this point show a linear increase and after the above mentioned point, the values of BAT increase more steadily. Thus indicating the fact that the BAT values become steadier when cell load is increased and saturation is achieved. Figure 8 also shows that analytical and simulation results present the same performance trend when offered load by the stations in the cell increases gradually.

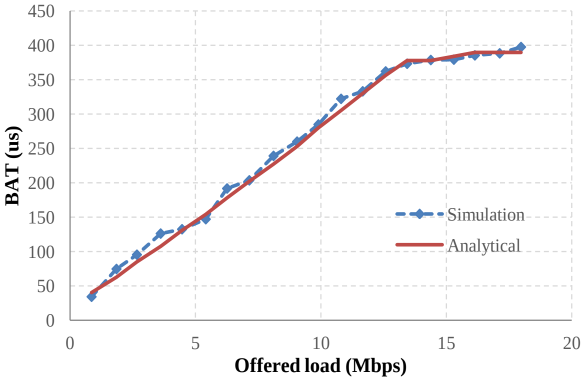


Figure 8. BAT values vs. offered load with 8 STAs at 18Mbps.

Finally, we also present BAT evaluation when multi-rate stations are used within a cell. Figure 9 analyzes a situation in which three stations are present and each of the stations uses the combinations of three physical transmission rates (i.e. 6, 18 and 54 Mbps). All of the three stations use the constant payload size of 1500 Bytes. It is evident that if all the stations use the smaller physical rate of 6 Mbps, the BAT values are the maximum. This is because all the stations would require more time to finish the transmissions. On the contrary, when all the stations use the maximum rate of 54 Mbps, the BAT value is reduced, further elaborating the above statement. Similarly, the pattern in the Figure 9 indicates the fact that, as more stations use higher physical rates, the BAT value decreases correspondingly. Figure 9 also indicates the accuracy of the exposed analytical model.

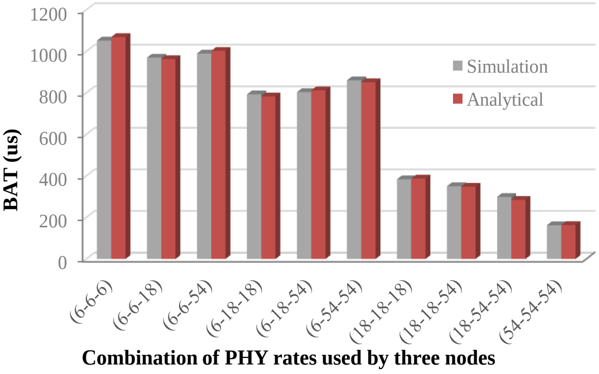


Figure 9. BAT values when stations with different transmission rates are used within a cell.

3.3.3. Testbed Measurements

BAT measurements were obtained from a PC running Linux and hostapd[5] to work as an AP through a Madwifi[6]-compatible Atheros-based WLAN NIC. The Madwifi driver was modified to provide instantaneous BAT samples to user-space. In fact, the modified driver provides the time taken from the moment at which the beacon is generated (i.e. TBTT) to the moment at which the driver receives an interruption reporting the transmission of the beacon. We define

[5] Host AP and WPA supplicant: http://w1.fi/hostapd/

[6] The MadWifi Project: http://madwifi-project.orgi

this statistic provided by the driver as BAT′ and is given by BAT′= BAT + transmission time of beacon + delay of the interruption. The transmission time of a beacon is known since, in this particular case, the AP broadcasts 116Byte-long beacon frames at 1Mbps. However, the delay of the hardware interruption is unpredictable and may depend on the load of the operating system. We measured this delay in the testbed depicted in Figure 3 with the AP running the minimum possible number of processes and under different Ethernet traffic load conditions. In those experiments we found that BAT′ was stable at 1126μs. Therefore BAT = PIFS + BAT′- 1126μs.

We measured BAT with the AP serving 1 to 3 saturated STAs, using three different PHY rates (6, 18 and 54Mbps) and three different frame sizes (40, 500 and 1500Bytes) generated by the *iperf* tool. Figure 10 compares measurement results with simulations and analytical values. For the sake of brevity, the figure shows averaged results of experiments with different number of STAs. The experiments showed that measurements differ significantly from both analytical values and simulations when the PHY rate of the STAs is low (error of 17% in the worst case). For moderate and fast modulations, BAT lays within the expected range. Overall, the average error is around 5%.

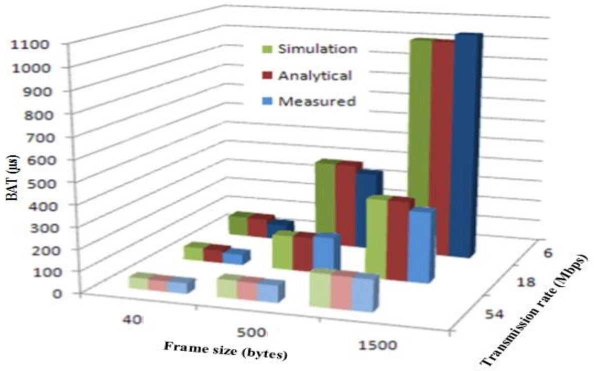


Figure 10. Measured, simulated and analytical BAT values using different transmission rates and different frame sizes.

4. Evaluation of a jammer and cheater BAT-based detector

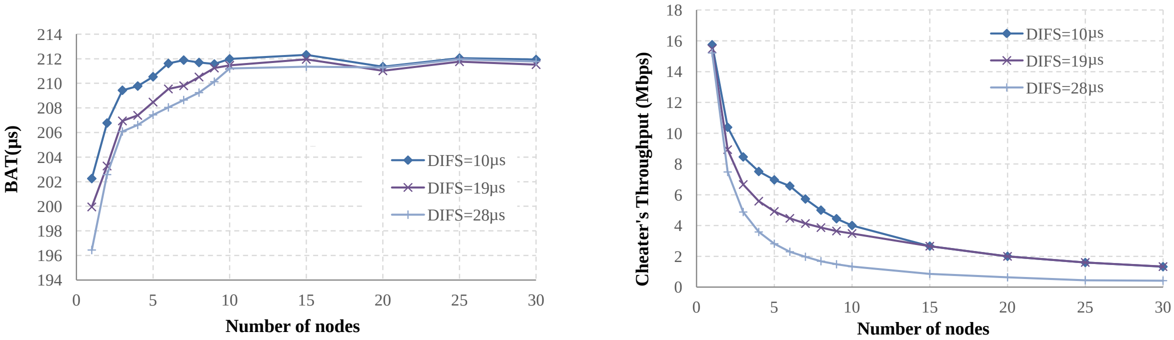
In the previous sections we have proved that BAT can be predicted by an AP under normal operation of the WLAN, provided that the statistics of the traffic are known. In this section we evaluate the effectiveness of a jammer detection mechanism based on BAT measurements. First, by means of simulations, we study how the presence of a jammer or a cheater STA makes BAT measurements deviate from the expected values, as predicted by the model presented in 3.2. Second, we implement the jammer detector and test its effectiveness in a small testbed. Recall that, with downlink traffic, BAT is kept constant and thus the presence of a jammer will easily be detected. Conversely, the worst case scenario (i.e., the most interesting scenario) is found when all traffic is uplink. For these reasons, our evaluation only considers uplink traffic.

4.1. Evaluation of BAT in the presence of a Cheater

In this section, we observe the impact of cheater in IEEE 802.11 networks and analyze its effect on the BAT value. Within this analysis we simulate scenarios where up to 30 stations are communicating to a single AP (on the uplink) with a constant physical transmission rate of 24 Mbps and a constant frame size of 1000Bytes. One of the stations acts as a cheater where it misuses the IEEE 802.11 MAC constraints in order to attain bandwidth gains. Also, the traffic offered by the stations is kept near saturation point.

4.1.1. Cheating device varying DIFS

In the normal operation of IEEE 802.11 MAC, the value of DIFS is set to the default value of 28μs. A cheating device can be able to attain the access of the shared channel much more quickly if it can reduce its DIFS value. In this way, the cheater can prioritize its communication as compared to other stations.



(a) Evolution of BAT values.

(b) Throughput of the cheater.

Figure 11. Simulation results showing BAT and throughput with the increase in number of nodes and the presence of a cheater varying its DIFS.

We simulate the scenarios where we vary the number of stations from 1 to 30. The smallest Inter frame space used in IEEE 802.11 is the SIFS and its default value is 10μs. The DIFS value for the cheater is increased from 10μs to 28μs. Figure 11a indicates that the BAT value decreases when the cheater's DIFS is increased because having a larger DIFS value reduces its priority to access the channel. The BAT value is also increased as the number of stations increase. However, when there are more than 10 STAs, the BAT value converges, making it difficult to detect the cheater. This is due to the fact that with more stations, the probability to find the channel busy when a TBTT arises is greater.

Figure 11b indicates the throughput obtained by the cheater, when it reduces it DIFS value from the default (i.e. 28 μs). The throughput of the cheater is increased when the DIFS value is decreased and is maximum when DIFS value is set to SIFS (i.e 10 μs). Despite its clear advantage over other stations, the cheater still needs to contend for the medium and its throughput will therefore decrease as the number of competing stations increases.

4.1.2. Cheating device varying Minimum Contention Window

Following a DIFS period, stations willing to transmit a frame will back off for a random number of time slots chosen between 0 and the value of the contention window CW. The default value of minimum CW, CW$_{min}$ is 16. A cheater can utilize lesser CW$_{min}$ value than 16 which can reduce its backoff and thus increase its access probability.

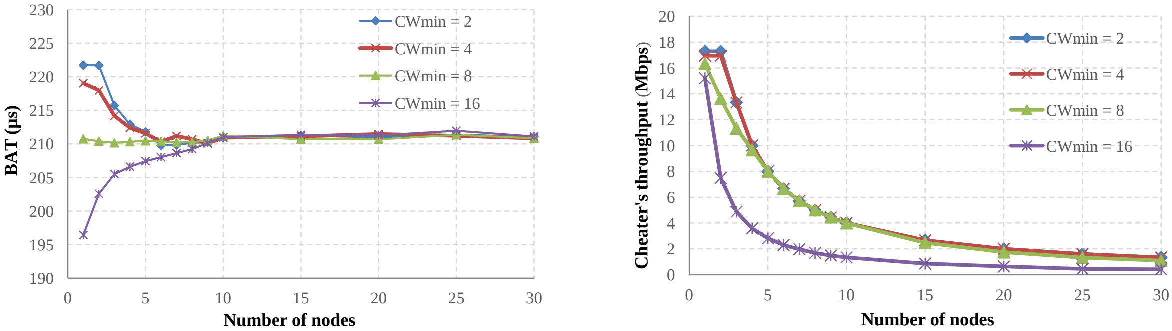
Figure 12a indicates the fact that the BAT value is greater when the cheater uses a lower CW$_{min}$ value. It is due to the fact that the cheater reduces its average backoff time after collision and is likely to gain the access to the shared medium much more quickly, as compared to the other stations. As in the case of reduced DIFS, the change in trend of BAT values is much more observant when less than 10 stations are communicating within a cell.

The throughput of the cheater is considerably increased with the reduced CW$_{min}$, as shown in Figure 12b. It is also notable that the throughput of the cheater decreases with an increase in the number of stations and converges to the performance observed when the cheater employs the default CW$_{min}$ value.

It is apparent from our simulation results that a cheater can cause detectable changes in the BAT value of the AP, provided that the number of active STAs is not very large (e.g. smaller than 10). This change in BAT value depends on the strategy being employed to achieve bandwidth gains.

4.1.3. Comparison between DIFS and CW-based cheating strategies

In our study, we investigated how a cheater can increase its throughput by changing the DIFS, CW$_{min}$ and CW$_{max}$ values. When we compare the cheating strategies used by the cheater with each other, it is pertaining to note that the change in BAT value is highest for the case when the cheater is employing a decreased CW$_{min}$. But this strategy also provides the cheater with highest throughput gain when compared to the other strategies. For the case of decreased DIFS, the BAT values are slightly raised but the throughput of the cheater is considerably increased. It is interesting to mention that a decrease in CW$_{min}$ strategy results in greatest BAT values at AP that can caution the AP for the presence of a cheating device. After comprehending the indicators, the AP can take necessary actions to limit the impact of the cheater. On the other hand, the decrease in DIFS strategy causes minor changes in BAT values that might not



(a) Evolution of BAT values.

(b) Throughput of the cheater.

Figure 12. Simulation results showing BAT and throughput with the increase in number of nodes and the presence of a cheater varying its CW$_{min}$.

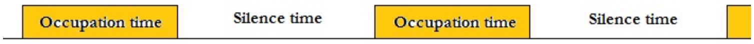


Figure 13. Characteristic diagram of On-Off jammer.

raise alarms at the AP; the cheater can go unnoticed, while still being able to attain greater share of bandwidth. The strategy where the cheater decreases its CW$_{max}$ was found to have minimum or no impact on the BAT value as well as the throughput of the cheater, thus indicating that this strategy is fruitless for both the cheater and the network.

4.2. Evaluation of BAT in the presence of a Jammer

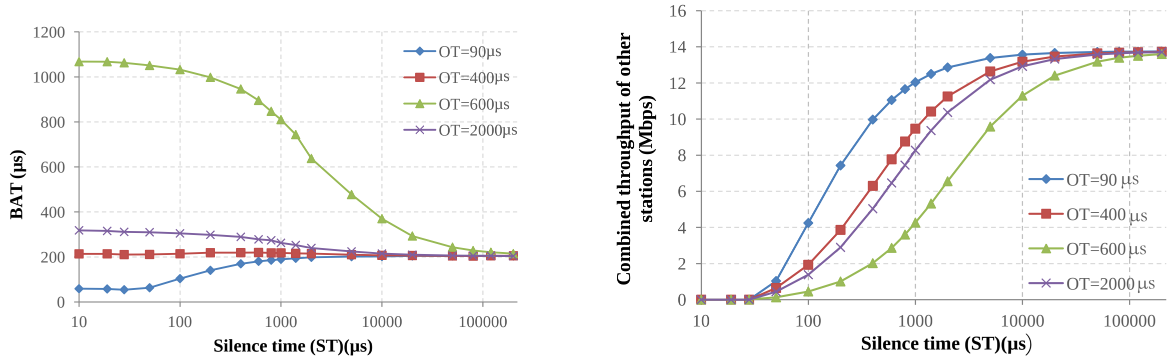
In order to understand the impact of jammers on BAT value, we simulate scenarios where one jammer injects false messages periodically. This jammer is placed randomly within the cell and is able to affect the transmission of both the AP and the STAs. Similar to [10], we use an On-Off jammer that jams/transmits continuously for a certain time (called Occupation time, or OT) and sleeps for a certain time (called Silence time, or ST), thus enabling a more thorough analysis than a simpler continuous jammer. We analyze the impact of the jammer by tuning its occupation and silence times. We simulate scenarios with 10 stations communicating to a single AP (on the uplink) at a constant physical transmission rate of 24 Mbps and a constant frame size of 1000 Bytes. The jammer is present in the vicinity of the AP and therefore interferes with both transmitters and receivers.

4.2.1. Variation in silence time

In order to understand the impact of silence time on the AP's BAT, we simulate cases where the silence time of the jammer is increased. In order to make our findings more concurrent, we utilize 4 different values for the occupation time. For a particular simulation scenario, the occupation time is kept constant, whereas the silence time between transmissions is increased from 10μs (i.e. SIFS) to 200ms.

Figure 14a illustrates the impact on BAT value when silence time of the jammer is increased. Logically, the BAT value is greater when the jammer occupation time is higher. As the silence time between transmissions is increased, the BAT values for all the combinations converge because the effect of the jammer is minimized. Increase in silence time allows channel to be more frequently available to legacy stations. Figure 14b depicts the effect on combined throughput of all legacy stations, in the presence of the jammer. It is also apparent that the throughput of the stations increases as the effect of the jammer is reduced. The combined throughput of all the stations for different jammer settings, converge to a single point when silence time becomes too large as to have any impact.

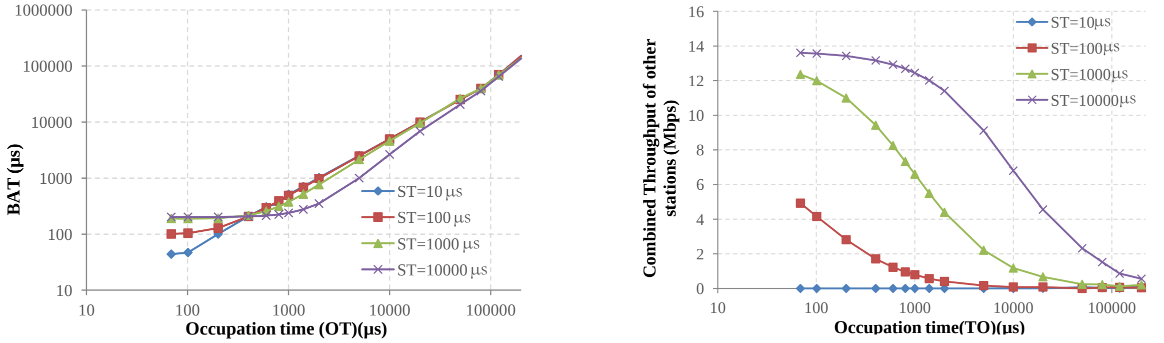
Figure 14a indicates that the least amount of variations in BAT value is for the case when 400μs of occupation time is used along with the varying silence time. That is, a jammer can remain un-noticed while utilizing the occupation time of 400μs, but still can be able to reduce the throughput of other legacy stations. This behavior is explained in the next section.



(a) Evolution of BAT vs. silence time (log scaled axis).

(b) Combined throughput of legacy stations vs. silence time.

Figure 14. Simulation results showing the effects on BAT and throughput in the presence of a jammer varying its silence time.



(a) Evolution of BAT vs. occupation time (log scaled axis).

(b) Combined throughput of legacy stations vs. occupation time.

Figure 15. Simulation results showing the effects on BAT and throughput in the presence of a jammer varying its occupation time.

4.2.2. Variation in occupation time

In order to understand the impact of occupation time over BAT values, in this simulation scenario we increase the occupation time of a jammer from 70μs to 200ms. We use 4 different constant silence times for each of the graphs plotted in Figure 15a. The trends indicate that, as the occupation time increases, BAT value is raised. At the occupation time of 400μs, the BAT values for all the trends converge. This is due to the fact that at 400μs occupation time the jammer mimics the behavior of a compliant station (i.e. sending 1000Byte frames at 24Mbps).

Figure 15b indicates that, as the occupation time for the jammer increases, the combined throughput of all compliant stations decreases. It decays slowly in the case where we have a greater silence period. This is because having a greater silence period gives time to other stations to communicate with the AP. At 200ms occupation time, the throughput of the combined stations approaches 0.

Analysis of Figure 15b indicates that, as the occupation time is increased, the impact of silence time is reduced. But it is interesting to mention the case where the silence time is set to be 100μs. Figure 15a shows that the BAT value for 100μs silence time follows the other trends. But Figure 15b shows that the combined throughput of legacy stations for the case of 100μs silence time is considerably reduced as compared to the silence time of 1000μs or 10000μs.

4.2.3. Tuning the most effective On-Off jammer

In order to compare the impacts of change in silence and occupation time on the jammer, we analyze Figures 14a, 14b, 15a and 15b. We indicate the situation where the jammer can be most deceptive. It is evident that, when the jammer utilizes an occupation time of 400μs and silence time of 100μs, it mimics the behavior of normal station (i.e. there is no considerable variations in the BAT value) and thus remains un-detected. Although these jammer settings do not completely prevent communications in the attacked WLAN, they reduce its capacity by a 85%.

Despite its apparent simplicity, an effective On-Off jammer that runs unnoticed to the BAT-based jammer detector must be aware of the frame length distribution of the attacked stations and try to mimic their behavior; this will actually require a rather sophisticated device. In conclusion, the BAT-based jammer detector will detect any type of jammer other than a well tuned reactive or intelligent jammer; for detecting the latter, it would be required that all STAs in the WLAN participate in the detection, whereas our approach lies completely on the AP.

4.3. Testing the effectiveness of BAT vs. PDR based detector

Throughout this section, we have shown how deviations in the measured BAT denote the presence of a malicious device and can thus be effectively used to trigger the jammer or cheater detection. In this subsection, we test the effectiveness of a simulated implementation of a BAT-based jammer detector and compare it against a PDR-based approach, which is the most common strategy in the literature (cf. Section 3). The main mode of detection employed in [2], [8], [9], [10] and [11] is based on the combination of RSS (or SINR) and PDR. PDR/RSS detection is based on the fact that, under normal conditions, the expected PDR is a function of the received signal quality (RSS or SINR). In these simulations, the baseline providing the expected PDR consists of two nearby static transmitters which are constantly sending (i.e. in saturation) 1000Byte frames at 24Mbps; their PDR without jamming is between 88% and 89%. We first consider a sensitive PDR-based jammer detector (S-PDR), a device that triggers the detection alarm when PDR is below 87%. We also consider a loose PDR-based detector (L-PDR), which detects a jammer when PDR goes below 80%. On the other hand, the simulated BAT-based predictor (BAT) raises an alarm when the measured BAT deviates more than 10% from the expected value.

Table 4 shows the effectiveness of the three detection strategies with '1' for false positives, '-1' for false negatives, and '0' for proper detection. Two sets of simulations with seven different cases each are evaluated. The first set of simulations is intended to measure false positive rates by introducing a hidden node, that is, a new non-jamming transmitter placed beyond the sensing range of the jammer detector but within the coverage of its receiver. The hidden node problem is common in relatively dense or dense Wi-Fi deployments. Different cases are evaluated where the hidden node progressively increases its offered load until saturation (case 7). The hidden node produces collisions that are translated into frame errors in the receiver and, therefore, triggers the jammer detection in the PDR-based mechanisms, while the BAT remains unaffected regardless of the hidden node activity.

Test case number:		1	2	3	4	5	6	7
Hidden node	BAT	0	0	0	0	0	0	0
	S-PDR	1	1	1	1	1	1	1
	L-PDR	0	0	0	1	1	1	1
Jammer	BAT	-1	-1	0	0	0	0	0
	S-PDR	0	0	0	0	0	0	0
	L-PDR	-1	-1	-1	0	0	0	0

Table 4. Successful detection rate of BAT and PDR-based strategies

The second set of simulations introduces an On-Off jammer with a fixed period of 2ms (duration of the longest frame at slowest rate) and whose duty cycle varies from 10% (case 1) to 90% (case 7). S-PDR strategy is able to successfully detect the jammer in all cases even though PDR is never drastically decreased; note that, during the On periods, CSMA stations will avoid collisions (i.e. losses) by deferring their transmissions upon detection of the energy transmitted by the jammer. This is misleading for L-PDR in the first cases. The threshold set for the PDR strategy clearly determines its effectiveness, however, we have to note that there was no configuration providing a perfect detection and, at the same time, a low false positive rate. Finally, even though BAT was affected in all cases, the first two produced deviations below 10% of the expected BAT; the jammer in those two cases was hardly noticeable in terms of capacity reduction.

4.4. Implementation of a BAT-based jammer detector

Our implementation of the BAT-based detector is a user-space daemon running in a Linux-based IEEE 802.11g AP and the Madwifi driver. The detector consists of four modules, as depicted in Figure 16. The basic operation

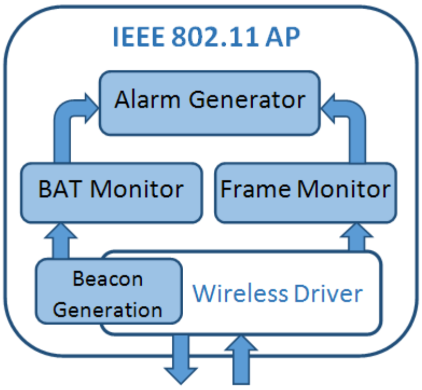


Figure 16. Modular description of BAT-based detector implementation.

of the detector is as follows. A Frame Monitor gathers Wi-Fi traffic statistics that are used to compute the expected average BAT. $\overline{BAT}$ predictions follow the analytical model presented in 3.2. T$_{data}$, P$_i$ and P$_{busy}$ are obtained from per-associated STA statistics made available by the Madwifi driver set in AP mode, which include the rate, the number of frames and the number of bytes transmitted to or received from all associated STAs. The Beacon Generator module generates beacon frames and triggers interruptions caught by the BAT Monitor module the instants before and after a beacon frame is transmitted. The BAT Monitor module provides the Alarm Generator module with the measured $\overline{BAT}$. Finally, the Alarm Generator module compares the measured and the predicted $\overline{BAT}$ and decides whether to generate a jammer alarm, according to the following design choices.

As detailed in section 3.3.3, we modified the Madwifi driver to provide BAT measurements after every beacon transmission. At the BAT Monitor module, those BAT samples are used to obtain the average BAT according to a moving average that considers the last s samples. Choosing the value of s poses a trade-off between reliability and timeliness; the higher the number of samples considered, the higher the $\overline{BAT}$ estimation accuracy; on the other hand, if a big value is chosen for s, it will take longer to detect the presence of a jammer/cheater.

Assuming that $\overline{BAT}$ is a random variable uniformly distributed between 19μs and 2138μs[7], $s = 120$ guarantees, with a confidence of 95%, that the estimated $\overline{BAT}$ has an error smaller than 10%. Using the default Beacon Interval of 100ms, 120 samples require 12s. That is, a jammer or cheater producing variations greater than 10% in $\overline{BAT}$ would be detected after 12s or less. Note that a reduced Beacon Interval will allow an improved accuracy of the estimation and a reduced detection time. Beacon interval can be reduced up to 25ms without significantly affecting the BSS throughput performance [22].

We also have to note that, in a noisy environment such as the 2.4GHz ISM band, BAT measurements are affected by other nearby transmissions. In order to minimize the number of false positives produced by non-malicious interference, the sensitivity of the detector must be tuned according to the normal activity of the channel. In our case, the detector report the presence of a jammer when the measured $\overline{BAT}$ exceeds the predicted $\overline{BAT}$ by 300μs. This threshold further reduces the effectiveness of the detector in the presence of discreet cheaters (cf. section 4.1).

The resulting application uses a small portion of the AP's memory (less than 30kB) and requires low CPU resources, making it suitable for running on commercial APs. The detector module has been tested under different traffic conditions (0 to 3 associated STAs with varying offered traffic, different frame sizes, etc.). In our tests, the activation of the jammer described in section 4.2 was always correctly detected at a maximum distance of 120m[8] (open office indoor environment), even before its presence could be noticeable on the carried throughput. No false alarms arose even though the tests were carried out during office hours (i.e. while the surrounding WLANs are heavily loaded).

[7] 19 and 2138μs correspond to the minimum and maximum possible values for BAT, being PIFS the minimum, and 2138μs the duration of the largest allowed frame using the slowest modulation of 6Mbps.

[8] Limit imposed by the dimensions of the facilities where the measurements took place.

5. Conclusions and future work

In this paper, we design, implement, and evaluate a BAT-based malicious detection scheme for IEEE 802.11 networks.

In order to better understand the threat that a jammer entails, we first analyzed the impact of On-Off and continuous jammers in IEEE 802.11 based WLAN networks by means of simulations and measurements utilizing an actual and publicly available jamming device. Then, we defined BAT as the metric upon which our proposed jammer and cheater detection scheme is built. The aforementioned scheme is based on the sheer principle that transmissions of beacons have priority over any other transmission and thus can be used to monitor the activity within WLAN network area from its APs (i.e. not requiring any modification to legacy client STAs). An analytical model is presented that is able to predict BAT. The effectiveness of our detection scheme is evaluated by simulating BAT over IEEE 802.11 based simulator as well as by experimentation where BAT is measured over a real IEEE 802.11 network.

The results indicate consistency among the simulation and the experimentation results and both of these are found to follow the trends indicated by the analytical model. Furthermore, simulation results show that our scheme is able to detect a cheating device as well as a jammer within the network. The detection process is more obvious when 10 or less stations are communicating with an AP. Our scheme detects more easily those cheaters that vary their CW_{min} than those with reduced DIFS.

Most of the handheld and low-priced jamming devices that are easily available today in the market can be classified as memoryless continuous or periodic (On-Off) jammers. Throughout our simulations and experiments we proved that these types of jammers are also detected by our proposed mechanism. However, intelligent jamming devices that mimic the behavior of a legitimate IEEE 802.11 STA could run unnoticed.

Measurements results indicate that our scheme can correctly detect a malicious entity at a distance of, at least, 120m. The BAT based detection scheme is overall found to be efficient in detecting a malicious entity within IEEE 802.11 based WLAN networks.

Although our study is based on IEEE 802.11a/g devices, our malicious entity detection scheme can be equally employed to newer IEEE 802.11 standards, such as IEEE 802.11n or IEEE 802.11ac, since they utilize the same CSMA/CA based MAC layer, which is the source of the weakness exploited by the jammers and cheaters of our study. As a future work, we plan to extend our study by exposing BAT based detecting technique to intelligent jammers. Furthermore, we would like to utilize the feedback information from the stations to further enhance the detection process. Also in future, we would like to perform analysis of cooperating APs to detect a malicious entity (by comparing BAT of each AP).

6. Acknowledgements

This work was partially supported by the Spanish Government through project TEC2012-32531 and the i2CAT Foundation.

References

[1] A. Fragkiadakis, I. Askoxylakis, P. Chatziadam, Denial-of-service attacks in wireless networks using off-the-shelf hardware, in: N. Streitz, P. Markopoulos (Eds.), Distributed, Ambient, and Pervasive Interactions, Vol. 8530 of Lecture Notes in Computer Science, Springer International Publishing, 2014, pp. 427–438.

[2] W. Xu, W. Trappe, Y. Zhang, T. Wood, The feasibility of launching and detecting jamming attacks in wireless networks, in: Proc. of ACM MOBIHOC, 2005, pp. 46–57.

[3] IEEE Standards Association, IEEE Standard for Information technology–Telecommunications and information exchange between systems Local and metropolitan area networks–Specific requirements Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications, ANSI/IEEE Std. 802.11-2012.

[4] K. Pelechrinis, M. Iliofotou, S. Krishnamurthy, Denial of Service Attacks in Wireless Networks: The Case of Jammers, IEEE Communications Surveys Tutorials 13 (2) (2011) 245–257.

[5] X. Zou, J. Deng, Detection of fabricated CTS packet attacks in wireless LANs., Springer Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering 74 (2010) 105–115.

[6] K. Pelechrinis, G. Yan, S. Eidenbenz, S. Krishnamurthy, Detecting Selfish Exploitation of Carrier Sensing in 802.11 Networks, in: Proc. of IEEE INFOCOM, 2009, pp. 657–665.

[7] E. Garcia-Villegas, M. Gomez, E. Lopez-Aguilera, J. Casademont, Detecting and mitigating the impact of wideband jammers in IEEE 802.11 WLANs, in: Proc. of IWCMC, 2010, pp. 57–61.

[8] K. Pelechrinis, I. Koutsopoulos, I. Broustis, S. Krishnamurthy, Lightweight Jammer Localization in Wireless Networks: System Design and Implementation, in: Proc. of IEEE GLOBECOM, 2009.

[9] Z. Liu, H. Liu, W. Xu, Y. Chen, Exploiting Jamming-Caused Neighbor Changes for Jammer Localization, IEEE Transactions on Parallel and Distributed Systems 23 (3) (2012) 547 –555.

[10] A. G. Fragkiadakis, V. A. Siris, N. E. Petroulakis, A. P. Traganitis, Anomaly-based intrusion detection of jamming attacks, local versus collaborative detection, Wireless Communications and Mobile Computing (2013).

[11] K. Ju, K. Chung, Jamming Attack Detection and Rate Adaptation Scheme for IEEE 802.11 Multi-hop Tactical Networks, International Journal of Security and Its Applications (2012) 149–154.

[12] K. Pelechrinis, I. Broustis, S. V. Krishnamurthy, C. Gkantsidis, A measurement driven, 802.11 anti-jamming system, CoRR abs/0906.3038.

[13] G. Noubir, R. Rajaraman, B. Sheng, B. Thapa, On the robustness of ieee 802.11 rate adaptation algorithms against smart jamming, in: Proceedings of the Fourth ACM Conference on Wireless Network Security, WiSec '11, ACM, New York, NY, USA, 2011, pp. 97–108.
URL `http://doi.acm.org/10.1145/1998412.1998430`

[14] L. Deek, E. Garcia-Villegas, E. Belding, S.-J. Lee, K. Almeroth, Joint rate and channel width adaptation for 802.11 MIMO wireless networks, in: Proc. of IEEE SECON, 2013, pp. 167–175.

[15] M. Acharya, T. Sharma, D. Thuente, D. Sizemore, Intelligent jamming in 802.11b wireless networks, in: Proc. of OPNETWORK, 2004.

[16] M. Raya, J.-P. Hubaux, I. Aad, Domino: A system to detect greedy behavior in ieee 802.11 hotspots, in: Proceedings of the 2Nd International Conference on Mobile Systems, Applications, and Services, MobiSys '04, ACM, New York, NY, USA, 2004, pp. 84–97.

[17] J. Tang, Y. Cheng, W. Zhuang, Real-time misbehavior detection in ieee 802.11-based wireless networks: An analytical approach, Mobile Computing, IEEE Transactions on 13 (1) (2014) 146–158.

[18] E. Garcia-Villegas, J. Ferrer, E. Lopez-Aguilera, R. Vidal, J. Paradells, Client-driven load balancing through association control in IEEE 802.11 WLANs, European Transactions on Telecommunications 20 (5) (2009) 494–507.

[19] G. Bianchi, Performance analysis of the IEEE 802.11 distributed coordination function, Selected Areas in Communications, IEEE Journal on 18 (3) (2000) 535–547.

[20] E. Lopez-Aguilera, J. Casademont, E. Garcia-Villegas, A study on the influence of transmission errors on WLAN IEEE 802.11 MAC performance, Wireless Communications and Mobile Computing 11 (10) (2011) 1376–1391.

[21] E. Lopez-Aguilera, J. Casademont, J. Cotrina, Propagation delay influence in IEEE 802.11 outdoor networks, Wireless Networks 16 (4) (2010) 1123–1142.

[22] E. Lopez-Aguilera, J. Casademont, J. Cotrina, IEEE 802.11g performance in presence of beacon control frames, in: Proc. of IEEE PIMRC, 2004, pp. 318–322.