Beacon Frame Spoofing Attack Detection in IEEE 802.11 Networks

Asier Martínez*, Urko Zurutuza†‡, Roberto Uribeetxeberria†, Miguel Fernández†,
Jesus lizarraga†, Ainhoa Serna† and Iñaki Vélez†
*Abit Security, Uribarri Etorbidea 19 - 1º
Polo de Innovación Garaia, 20500 Mondragon, Spain
Tel.: +34 943 712 072
Email: amartinez@sakontek.es
†Mondragon University, Computer Science Department
Loramendi 4, 20500, Mondragon, Spain
Tel.: +34 943 739 634. Fax: +34 943 791 536
Email: uzurutuza@eps.mondragon.edu,
ruribeetxeberria@eps.mondragon.edu, mfernandez@eps.mondragon.edu,
jlizarraga@eps.mondragon.edu, aserna@eps.mondragon.edu,
ivelez@eps.mondragon.edu
‡This author is supported by the grant BFI05.454
of the Department of Research, Education and Universities
of the Basque Government.

***Abstract*—A great variety of well-known attacks exist for the IEEE 802.11 protocol. The lack of mechanisms for management frame authentication and the complexity of the protocol itself have derived into a considerable number of denial of service and identity spoofing attacks. As most denial of service attacks are based on spoofing of MAC addresses, spoofed frame detection schemes have gained attentions. Currently the most efficient techniques to detect this kind of attacks are based on the creation of profiles for the wireless nodes and behavior based protocol anomaly detection. However, these techniques tend to generate too many of false positives. This is caused by the unstable nature of the wireless medium and also because of the difficulty to model the behaviour of the diverse implementations from different manufacturers. One way to reduce false positives is to combine different techniques to carry out the analysis. We propose a novel method that identifies the impersonation of certain management frames, which helps to reduce the number of false positives within other existing MAC spoofing detection techniques.**

***Index Terms*—802.11 MAC address spoofing, false positive reduction, synchronisation attack detection, wireless intrusion**

I. Introduction

Wireless networks have gained much popularity lately, to such an extent that we can find them in almost any aspect of our daily life. Mobile phones, PDAs and computers are some evident examples. The most popular implementation for local area networks is the standard IEEE 802.11, also known as Wi-Fi. As Wi-Fi networks proliferated, the security flaws of the protocol became notorious.

Management frames carry out critical tasks in those networks, but unfortunately these frames are not authenticated. This is probably the most important weakness of the protocol. As a consequence, several denial of service (DoS) attacks are possible [1]–[3]. 802.11i and 802.1X standards have mitigated the effects of this problem but not all the possible attacks have been tackled and even worse, new ones have arisen [4]. Therefore it is necessary to develop techniques that will allow us to detect DoS attacks in 802.11 networks. Most of these attacks impersonate MAC frames, thus the detection of such impersonation could lead us to the detection of a great variety of attacks.

In this work we propose a new technique to detect the falsification of management frames in IEEE 802.11 protocol. More precisely, we give details about how to detect beacon frame falsification. These frames are responsible of distributing critical information in an 802.11 network. We propose an algorithm that identifies each false beacon frame in order to detect DoS attacks in a passive mode. The article contributes as follows:

- We describe beacon frame based attacks.
- We develop a method for a false positive-free, single false beacon frame detection.
- We show experimental results, analysis and a benchmark of our system implementation compared with a known IEEE 802.11 based intrusion detection system.

The rest of the document is organised as follows: Section II gives an overview of MAC address spoofing detection techniques. It focuses on the strong and weak points of each technique. Section III-A describes DoS attacks based on desynchronisation of nodes. These attacks are carried out by the impersonation of beacon frames. A method to detect these spoofed frames is proposed in section IV. After a theoretical

description of our detection method, section V shows the results of experimental tests over two different scenarios. Finally, conclusions extracted from the experimental work are detailed and summarised in section VI.

II. Related work

Despite the existence of diverse methods to detect the MAC frame spoofing in 802.11, widely all of them can be classified into two categories: protocol anomaly detection and anomaly detection based in the individual characteristics of 802.11 nodes.

Techniques belonging to the first category try to model and understand the normal behaviour of a 802.11 network. After modeling this behaviour, the network is monitored looking for patterns that do not fit into this model. One of the most popular techniques within this category uses sequence number analysis of 802.11 frames. This number acts as a sequence number identifier of the frames transmitted from a node. In this sense Joshua Wright proposes in [5] the use of this sequence number field in the frame. This is a very simple technique that uses a threshold representing the maximum difference between each sequence number. The main disadvantage of this approach is the amount of false positives generated. This happens because the theoretical model on which it is based does not properly fit the real operation of a 802.11 network [6]. Nevertheless this technique has been implemented in some free intrusion detection tools such as Snort-Wireless[1], WIDZ[2] or Garuda[3]. On the other hand, Fanglu Guo et al. [7] model the behaviour of the sequence field using an empirical method that takes measures in a 802.11 network for a given time. Although this method achieves a more realistic model, it can vary on for different devices [2], [6] or situations other than those used when taking the measures.

Also making use of the sequence number field, Dasgupta et al. [8] propose more precisely fuzzy logic techniques, to obtain more flexible patterns with a lower false positive rate. However, results obtained on tests have not been very encouraging. LaRoche et al. [9] use machine-learning techniques to model the behaviour of the protocol and reduce the number of false positives. Genetic algorithms are used in this work but the false positive ratio obtained does not offer a significant improvement.

Still within protocol anomaly detection, indirect detection is another approach to detect spoofed frames. Bellardo et al. describe an heuristic technique to detect de-authentication attacks in [2]. This kind of attack performs MAC address spoofing and therefore the attack can be detected indirectly.

Kismet[4] is a well-known 802.11 network scanner that includes intrusion detection features. It is able to model the behaviour of beacon frames and the detection of spoofed frames is based on the coherence of the *BSSTimestamp* field. This approach has obtained good results so far. *BSSTimestamp* field consists on a counter of the time (in microseconds) that the access point is active. For example, if the *BSSTimestamp* does not increase with time, the value will not be coherent and an anomaly will be detected. In practice, modeling the behaviour of a 802.11 network is not a simple task. The unstable nature of the wireless medium and the different implementations of the protocol in network cards [6] create important deviations between the behaviour of different networks [10], [11]. Nevertheless, although getting a general model for every attack seems impossible, sufficiently reliable and useful patterns can be obtained.

The creation of profiles with the characteristics of wireless nodes is an alternative to protocol modelling. These profiles are created using measurable attributes of each wireless node. Characteristics such as hardware [12], [13], software [14] and firmware [15], [16] fingerprints analysing the behaviour of the node could be included. Also attributes referring to the physical position of the node can be used. In [17], [18] the delay in the transmission of fixed length frames and the fluctuation of the power in the received signal is used to univocally identify each node. More simply, in [19]–[21] the validity of the physical addresses of MAC frames is verified. Unfortunately this will only detect the spoofing of non-existent nodes and it would be very simple to overcome by generating valid addresses and thus remain undetected.

III. Beacon based attacks

A. Synchronization attacks

A beacon frame is used for several functions. To synchronise the clocks of the nodes and to announce the existence of the network as well as to transmit some necessary configuration parameters to join it [22]. Other important functions of beacon frames are related to the maintenance of the network. Beacon frames are transmitted at regular intervals to allow the nodes find and identify a network. Every wireless network needs a coordinator in charge of transmitting beacon frames.

1) Power Saving Mode Attack: PSM allows nodes to save energy while they are waiting for the channel to be available for transmission. For example, one node will go to a power save mode for a period specified by the access point. During this idle time, the access point will buffer the packets destined to that node and they will be sent to it when it wakes up. If for any reason, the node wakes up at any other time than that expected by the access point due to desynchronization caused by spoofed beacon frames, it may loose the buffered information. As a result, the victim node can suffer a reduction in its capacity for transmitting [3].

2) PCF attack: In a PCF (Point Coordination Function) mode, the access point serves as a network referee. It provides the priority mechanisms for the devices. An attacker could spoof beacon frames using false clock values. Those values would produce a maladjustment in the contention periods of the stations, causing a DoS [3].

[1] http://www.snort-wireless.org/
[2] http://www.loud-fat-bloke.co.uk/tools.html
[3] http://sourceforge.net/projects/garuda/
[4] http://www.kismetwireless.net/

B. 802.11i attacks

The 802.11i standard is also propitious to suffering from attacks by means of the information contained in the beacon frames, as described in [23]. A manipulation of the element of network information of robust security specified in 802.11i will produce a DoS in the client node, keeping it from joining the network. If, for some reason, incoherence is detected in the security method chosen, the network joining process is aborted. This incoherence can be caused by an attacker who forges a beacon frame.

The rollback attack also exists, which tries to supplant negotiated values by the station by weaker encryption methods. [24] describe how to use the policies to detect this type of attacks, but it is not possible to detect the poisoning attack due to the fact that it modifies some bits that are insignificant and variable, causing the DoS without influencing the bits in charge of encryption or authentication.

C. False Information attacks

As previously described attacks do, false information attacks transmit manipulated values in the fields necessary for the stations to connect to the network. An example of this type of attack can be found in the WVE-2006-0050[5] wireless vulnerability database. The information field provides the number of the channels used by the network. If beacon frames are falsified using a wrong channel number, stations will not be able to join the network.

IV. Proposed detection method

The simplest way to detect most of the spoofed traffic is to modify the firmware of the access points and 802.11 cards in order to log the transmitted data. Knowing which frames have been transmitted helps to detect others that do not belong to the device even if they have the same physical address. This technique is very useful in infrastructure networks as the management frames are centralised in the access point. However, certain limitations exist in the market. On the one hand, the technique needs hardware with special firmware. On the other hand, it has to be taken into account that a lot of hardware without spoofing detection functionality already exist. External monitoring methods can help to overcome this necessity. They should be passive methods because of the lack of bandwidth that characterises wireless networks.

The technique proposed in this work detects beacon frames that have been spoofed in an infrastructure 802.11 network. This is a passive technique that does not need a modification of the firmware of the existing hardware. We have implemented it in a dedicated monitoring sensor. Spoofing of beacon frames can cause denial of service attacks as the ones mentioned in section III-A.

As said before beacon frames must be transmitted at regular intervals. This interval is specified by the access point and it is announced to the rest of the nodes in the "beacon interval" field. If a frame does not satisfy this condition, it can be considered as malicious. Nevertheless, exceptions for this behaviour exist. If a network is congested, the access point may delay the transmission of the beacon frame. This behaviour is not specified in the standard and using smaller beacon frame periods could be considered as a Hardware error, since an incorrect synchronisation may cause failure of some services. Therefore, the proposed technique is based on the monitoring of time intervals between beacon frames. In this work, we measure this value for each beacon frame transmitted and we define a variable called *Delta* which represents the time gap between two consecutive beacon frames. If Delta is smaller than a defined threshold, they will be considered as anomalous.

[5]http://www.wirelessve.org/entries/show/WVE-2006-0050

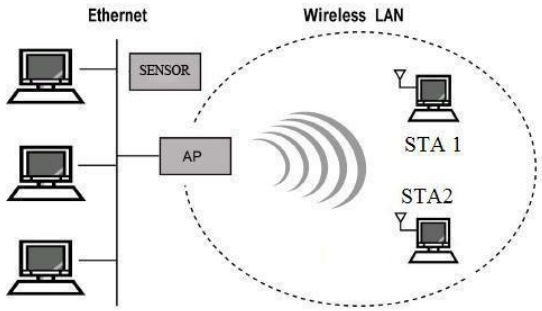


Fig. 1. Network diagram of test scenarios.

V. Experimental results

To test the validity of the new method proposed in section IV, the intrusion detection system for Wi-Fi networks Snort-Wireless has been modified. To measure the interval between beacon frames, the *MACTime* field of Prism [25] headers has been used. This field informs about the moment, in microseconds, when the wireless card received and stored the beacon frame. A more precise measure can be obtained as a result rather than simply analysing the time at the host. Two different scenarios have been created to complete the tests. This was because in practise the beacon frame intervals vary depending on the network traffic. The tests in the scenario of section V-B were made under low traffic conditions and the traffic was incremented for the scenario of section V-C.

A. Network configuration

Figure 1 shows the network configuration used during the experiments. There are two nodes with Senao 802.11g wireless cards generating traffic and a Linksys WRT54G access point operating in dual mode 802.11 b/g. The wireless sensor is located very close to the access point so the measurement of frame transmission times is very precise. The access point was configured with an interval between transmitted beacon frames of 102.4 ms.

B. Scenario I

In this first scenario, nodes generate moderate traffic by making Internet requests and SSH connexions. The attack was

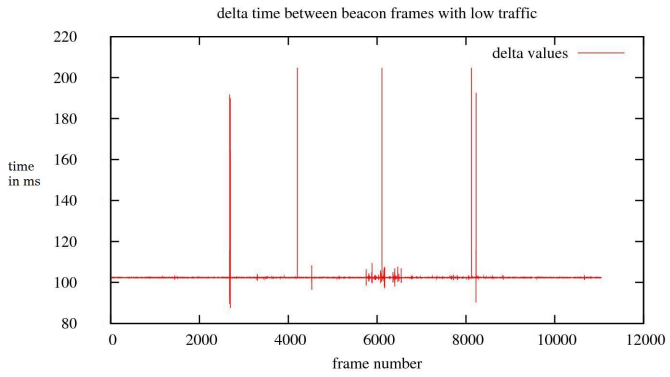
(a)

Threshold	FP	FN
1%	5	0
2%	0	0
3%	0	0
6%	0	0
10%	0	0

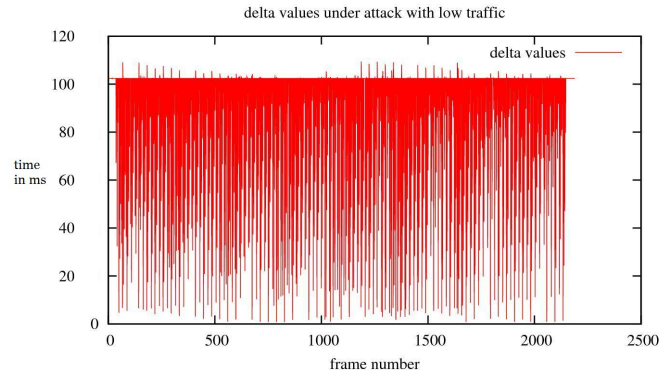
(b)

Threshold	FP	FN
1%	118	0
2%	4	0
3%	2	0
4%	1	0
5%	1	0
6%	0	0

TABLE I
FALSE POSITIVES AND NEGATIVES, A) IN A LOW TRAFFIC NETWORK DURING AN ATTACK B) IN A HIGH TRAFFIC NETWORK DURING AN ATTACK



(a)



(b)

Fig. 2. beacon frame delta times in a low traffic network.

carried out using a traffic injection tool called Scapy[6]. The tool sends three beacon packets per second after waiting a pseudo-random time obtained by the *random()* function (from the Python[7] programming language). Table 1(a) shows how the access point acts as expected and there is almost no divergence between the beacon frame *Delta* times.

1) Results for scenario I: Significant results have been obtained after carrying out the attacks. The mean value of delta time is considerably lower, the amount of false positive goes down rapidly and there are no false negatives. The absence of false negatives is due to the way that the attack was carried out. Desynchronization attacks need various frames to have

[6]http://www.secdev.org/projects/scapy/
[7]www.python.org/

(a)

Delta max.	204.808 ms
Delta min.	875.06 ms
Delta mean	102.451 ms
Delta variation	0.05%

(b)

Attack frame number	501
Attack frame loss	2
Delta min.	0.804 ms
Delta max.	109.376 ms
Delta mean	88.917 ms
Delta variation	13.16%

TABLE II
DELTA TIME STATISTICS IN A LOW TRAFFIC NETWORK. (A) DURING THE NORMAL OPERATION. (B) DURING AN ATTACK.

(a)

Delta max.	206.220 ms
Delta min.	96.639 ms
Delta mean	102.524 ms
Delta variation	0.122%

(b)

Attack frame number	501
Attack frame loss	29
Delta min.	0.826 ms
Delta max.	203.909 ms
Delta mean	89.615 ms
Delta variation	12.48%

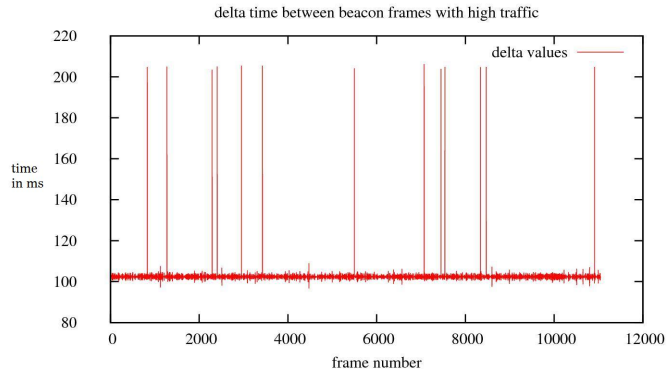
TABLE III
DELTA TIME STATISTICS IN A HIGH TRAFFIC NETWORK. (A) DURING THE NORMAL OPERATION. (B) DURING AN ATTACK.

some effect over the clocks of the nodes. In order to have a false negative, the absence of beacon frames should last for at least the double of the beacon interval predefined in the access point.

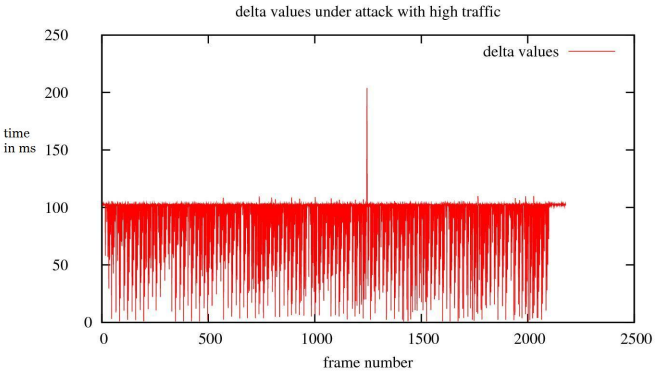
The difference between measured delta times of the beacon frames can be observed in figure 2. Figure 2(a) shows the values for a normal network while figure 2(b) shows how delta times decrease when a large amount of external beacon frames are introduced. In this case the predefined values are not kept anymore. It has to be mentioned too that the amount of false positives is very low. Table 1 shows how despite having a very low Threshold there are only five false positives. The reason for this is that the traffic is very low. Thus, intervals between beacon frames do not oscillate as much and they can be considered very precise. This can be compared with the results obtained in section V-C.

C. Scenario II

This scenario keeps save the previous network configuration. The difference only lays in the amount of traffic generated. Both client nodes make simultaneous transmissions of large files via FTP transferences. Due to this change, more fluctuations occur and are reflected in the statistics of table 3(b).



(a)



(b)

Fig. 3. Beacon frame delta times in a high traffic network.

1) Results for scenario II: Results of scenario II differ from those of scenario I because having higher traffic makes fluctuations between beacon frames grow. This is shown in table 2(b), where for a threshold of 1% 118 false positives are obtained while for scenario I there were only 5 of them. Statistics of the behaviour for a network that is not under attack also change. The deviation in a congested network is doubled as can be seen in table 3(a).As mentioned in section V-C, for high traffic in the 802.11 network, the hardware finds more difficulties to achieve the established beacon intervals. Therefore small fluctuations are generated and a high false positive rate will be produced if a low Threshold is established. This situation may change depending on the chosen hardware so the needed threshold will also be different required on the network electronics.

D. Trying to evade detection

Another significant result from the statistics shown in table 3(b) is that the value of delta time goes up to 203.9 ms (the maximum value measured) at least once. If an attacker was able to synchronise and inject the spoofed frame in a moment significantly close to the middle of the interval, he would manage to generate a false negative. Nevertheless it is not possible for an attacker to a priori know when those fluctuations will occur, and the congestion which caused the

(a)

Attack frames	499
Alerts	121
True Positives	90
False Positives	31
False Negatives	378

(b)

Attack frames	472
Alerts	110
True Positives	83
False Positives	27
False Negatives	362

TABLE IV
SNORT-WIRELESS ALERT RESULTS (A) DURING ATTACK WITH LOW TRAFFIC. (B) DURING AN ATTACK WITH HIGH TRAFFIC.

delay in the network will cause the invalidation of the attacker injected frame. In order to verify this, we suppose that an attacker can obtain a delay pattern. Practical attempts have been made, but it has been impossible to reproduce the attack due to the slow response times at the moment of injecting the frame resulting in the detection of the attack. These response times are much smaller than the required times. In addition, the fact that the machine, from which the traffic injection is made, does not have an operating system in real time causes that the synchronization of the attack became a complicated task. On the other hand, an attacker could try to interfere the legitimate frame and inject his own. Anyway this is not an easy task either [26] as wireless 802.11 networks make use of Direct Sequence Spread Spectrum (DSSS) which is very resistant against interferences. In addition to that, this kind of attack would require a highly specialised hardware and a correct synchronisation with the legitimate frame that we try to interfere with.

E. Comparison against Snort-Wireless

Snort-Wireless is the most advanced Open Source Wireless IDS. It uses the sequence number analysis technique proposed by [5] to detect false frame attacks. In this section we test the effectiveness of the Snort-Wireless with the used data applying the purposed analysis technique. Slightlyly modified default values have been used in Snort-Wireless to send out alerts in the detected attacks. This is because by default it only detects the first attack, saving the address of the attacker station without sending any alert in a period of time. Snort-Wireless is outdated in some aspects, but choosing Snort-Wireless instead of other commercial tools was due to the fact that they are a black box and it is imposible to analyze the techniques they use and to reach any satisfactory conclusion.

As shown in table 4, there is little difference between both high- and low-traffic scenarios. This happens because the traffic volume does not influence the behavior of the sequence number of the stations involved. It can also be observed that the detection rate is considerably lower. Even if spoofing attacks can be detected, it is not capable of identifying the malicious packets as the threshold-based technique used by Snort-Wireless is prone to false positives.

VI. CONCLUSIONS AND FURTHER WORK

The MAC address spoofing detection technique proposed in this article does not generate any false positive if correct detection threshold is established. Results clearly show that spoofed

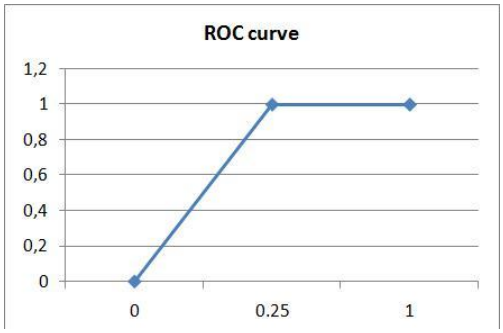


Fig. 4. ROC curve of the detection method in worst case with hight traffic.

beacon frames can be detected measuring the intervals between beacon frames. This method has revealed to be adequate to be implemented together with other techniques such as sequence number analysis. As well as being an effective technique its implementation is very simple a passive measurement with minimum hardware requirements is sufficient. Almost any 802.11 card could be used for that. This technique implies taking an step forward towards the creation of valid profiles that will allow us to detect anomalies in Wi-Fi networks. The introduction of spoofed frames in these networks generate anomalous situations. One of these anomalies can be caused by: not satisfying the minimum required intervals between frames, or other time intervals specified by the medium access control mechanisms of the protocol. The times can be measured and thus, the very same techniques can be used in the future to detect the anomalous behaviour provoked by other type of denial of service attacks. Although these techniques are not sufficiently strong to offer a fully reliable response by their own the reduction of false positives by means of combining this technique with other ones is possible. Finally, this technique could effectively be implemented in Ad-Hoc networks as they also use management frames that can suffer from the same kind of attacks.

REFERENCES

[1] W. A. Arbaugh, N. Shankar, and Y. J. Wan, "Your 802.11 wireless network has no clothes," *Wireless Communications, IEEE*, vol. 9, no. 1, pp. 44–51, 2002.

[2] J. Bellardo and S. Savage, "802.11 denial-of-service attacks: Real vulnerabilities and practical solutions," in *Proceedings of the Twelfth USENIX Security Symposium*. Washington, DC, USA: USENIX Association, Aug 2003, pp. 15–28. [Online]. Available: http://www.cs.ucsd.edu/~savage/papers/UsenixSec03.pdf

[3] G. Khanna, A. Masood, and C. Nita-Rotaru, "Synchronization attacks against 802.11," in *The 12th Annual Network and Distributed System Security Symposium Pre-Conference Wireless and Mobile Security Workshop*, San Diego, CA, USA, Feb. 2005. [Online]. Available: http://www.isoc.org/isoc/conferences/ndss/05/workshop/khanna.pdf

[4] A. Mishra and W. A. Arbaugh, "An initial security analysis of the ieee 802.1x standard," University of Maryland, Tech. Rep. CS-TR-4328, Feb 2002, uMIACS-TR-2002-10. [Online]. Available: citeseer.ist.psu.edu/566520.html

[5] J. Wright, "Detecting wireless LAN MAC address spoofing," http://home.jwu.edu/jwright/, 2003.

[6] A. D. Stefano, A. Scaglione, G. Terrazzino, I. Tinnirello, V. Ammirata, L. Scalia, G. Bianchi, and C. Giaconia, "Wifi does not imply 802.11 standard compliancy: experimental results," in *The Wireless Internet conference (WICON)*, July 2004.

[7] F. Guo and T. cker Chiueh, "Sequence number-based MAC address spoof detection." in *Proceedings of the 9th international symposium on recent advances on intrusion detection, RAID*, 2005, pp. 309–329.

[8] D. D. F. G. K. Yallapu and M. Kaniganti, "Multilevel monitoring and detection systems (MMDS)," in *Proceedings of the 15th Annual Computer Security Incident Handling Conference (FIRST)*, Ottawa, Canada, June 22-27 2003.

[9] P. LaRoche and A. N. Zincir-Heywood, "802.11 network intrusion detection using genetic programming," in *Genetic and Evolutionary Computation Conference (GECCO2005) workshop program*. Washington, D.C., USA: ACM Press, 25-29 Jun. 2005, pp. 170–171. [Online]. Available: http://www.cs.bham.ac.uk/~wbl/biblio/gecco2005wks/papers/0170.pdf

[10] J. Yeo, M. Youssef, and A. Agrawala, "A framework for wireless LAN monitoring and its applications," in *WiSe '04: Proceedings of the 2004 ACM workshop on Wireless security*. New York, NY, USA: ACM Press, 2004, pp. 70–79.

[11] A. A. Cárdenas, S. Radosavac, and J. S. Baras, "Detection and prevention of MAC layer misbehavior in ad hoc networks." in *SASN*, S. Setia and V. Swarup, Eds. ACM, 2004, pp. 17–22.

[12] J. Hall, M. Barbeau, and E. Kranakis, "Radio frequency fingerprinting for intrusion detection in wireless networks," *IEEE Transactions on Dependable and Secure Computing*, 2005.

[13] ——, "Enhancing intrusion detection in wireless networks using radio frequency fingerprinting," in *Ommunications, Internet and Information Technology (CIIT)*, November 2004, pp. 22–24.

[14] B. Sieka, "Active fingerprinting of 802.11 devices by timing analysis," in *Consumer Communications and Networking Conference,CCNC 2006*, 2006, pp. 15–19, Volume: 1.

[15] J. P. Ellch, "Fingerprinting 802.11 devices," Master's thesis, Naval Postgraduate School Monterey, California, 2006.

[16] J. Franklin, D. McCoy, P. Tabriz, V. Neagoe, J. V. Randwyk, and D. Sicker, "Passive data link layer 802.11 wireless device driver fingerprinting," in *Proceedings of the 15th USENIX Security Symposium*, Vancouver, Canada, jul-aug 2006, pp. 167–178. [Online]. Available: http://www.cs.cmu.edu/~jfrankli/usenixsec06/usenixsec06driverfingerprinting.pdf

[17] G. R. anbd Smith J., L. M., and A. . Clark, "Passive techniques for detecting session hijacking attacks in IEEE 802.11 wireless networks," in *Proceedings of AusCERT Asia Pacific Information Technology Security Conference (AusCERT2005)*, R. Stream, C. A., K. K., and U. o. Q. Mohay, G., Eds., 2005, pp. 26–38.

[18] R. Gill, J. Smith, and A. Clark, "Experiences in passively detecting session hijacking attacks in IEEE 802.11 networks," in *Fourth Australasian Information Security Workshop (Network Security) (AISW 2006)*, R. Safavi-Naini, C. Steketee, and W. Susilo, Eds., vol. 54. Hobart, Australia: ACS, 2006, pp. 221–230.

[19] W.-C. Hsieh, C.-C. Lo, J.-C. Lee, and L.-T. Huang, "The implementation of a proactive wireless intrusion detection system," in *CIT*, 2004, pp. 581–586. [Online]. Available: http://csdl.computer.org/comp/proceedings/cit/2004/2216/00/22160581abs.htm

[20] V. Sharma, "Intrusion detection in infrastructure wireless LANs," *Bell Labs Technical Journal*, vol. 8, no. 4, pp. 115–119, 2004. [Online]. Available: http://dx.doi.org/10.1002/bltj.10090

[21] Y.-X. Lim, T. Schmoyer, J. G. Levine, and H. L. Owen, "Wireless intrusion detection and response." in *IAW*, 2003, pp. 68–75.

[22] IEEE, "1999 edition (r2003) part 11: Wireless LAN medium access control (MAC) and physical layer (phy) specifications," IEEE, Tech. Rep., 1999 (R2003).

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

[24] R. Gill, J. Smith, and A. Clark, "Specification-based intrusion detection in WLANs," in *22nd Annual Computer Security Applications Conference*, December 11-15 2006.

[25] Intersil, *PRISM Driver Programmers Manual, version 2.30*, 2002, available at http://home.eunet.cz/jt/wifi/RM0251.pdf.

[26] W. Xu, W. Trappe, Y. Zhang, and T. Wood, "The feasibility of launching and detecting jamming attacks in wireless networks," in *MobiHoc '05: Proceedings of the 6th ACM international symposium on Mobile ad hoc networking and computing*. New York, NY, USA: ACM Press, 2005, pp. 46–57.