An automatic and self-adaptive multi-layer data fusion system for WiFi attack detection

An Automatic and Self-Adaptive Multi-Layer Data Fusion System for WiFi Attack Detection

Francisco J. Aparicio-Navarro, Konstantinos G. Kyriakopoulos, David J. Parish
School of Electronic, Electrical and System Engineering
Loughborough University
Loughborough, LE11 3TU, U.K.
e-mail: {elfja2, elkk, d.j.parish}@lboro.ac.uk

Abstract—**Wireless networks are increasingly becoming susceptible to more sophisticated threats. An attacker may spoof the identity of legitimate users before implementing more serious attacks. Most of the current Intrusion Detection Systems (IDSs) that employ multi-layer approach to help towards mitigating network attacks, offer higher detection accuracy rate and low numbers of false alarms than IDSs that employ single layer approach. However, few of the current multi-layer IDSs could be used off-the-shelf without a prior thorough training with completely clean datasets or fine tuning period. Dempster-Shafer theory has been used with the purpose of combining beliefs of different metric measurements across multiple layers. However, an important step to be investigated remains open; this is to find an automatic and self-adaptive process of Basic Probability Assignment (BPA). This paper describes a novel BPA methodology able to automatically adapt its detection capabilities to the current measured characteristics, with a light weight process of generating a baseline profile of normal utilisation and without intervention from the IDS administrator. We have developed a multi-layer based application able to classify individual network frames as normal or malicious with very high accuracy.**

Keywords-Basic probability assignment; Data fusion; Dempster-Shafer; Multi-layer measurements; Spoofing attacks; WiFi

I. INTRODUCTION

MAC layer spoofing attacks are among the most serious threats to wireless networks [1]. There exist numerous attacks, ranging from Denial-of-Service (DoS) to session hijacking that can be implemented because an attacker may masquerade itself as a legal user [2]. In the last few years there has been increasing interest in using detection methodologies to identify spoofing attacks in IEEE 802.11 networks [2, 9, 10]. The implementation of wireless network monitoring tools, such as Intrusion Detection Systems (IDSs), is fundamental in security infrastructures in order to provide another level of defence for IEEE 802.11 networks.

Although there are cases in which an algorithm that utilises a single metric approach might give positive results, the true status of a network is rarely accurately detectable by examining a single metric from one network layer of the protocol stack. As many researchers have previously demonstrated [3-5], the combined use of multiple metrics from the same or different network layers may result in higher detection accuracy rate with lower numbers of False Negatives (FN) and False Positive (FP). Hence, utilising a multi-layer approach may help towards automating the overall process of detecting and mitigating wireless network attacks.

Data fusion can be defined as the process of collecting information from multiple and heterogeneous sources, and combining them towards obtaining a more accurate final result [5]. The Dempster-Shafer (D-S) theory of evidence is a good candidate for this purpose. D-S has been previously used in the intrusion detection field to enhance the detection accuracy [5-7].

Despite having been proven as a powerful and efficient technique, a very important step to be investigated remains open in D-S theory. This is to find an automatic and self-adaptive process of Basic Probability Assignment (BPA), based on the measured characteristics of the network. The major challenge for applying D-S theory in IDS is to automatically determine the beliefs from the network measurements [8].

There exist multiple ways of assigning probabilities to each of the hypotheses in D-S theory, ranging from data mining techniques to empirical approaches. However, few of them could be used off-the-shelf without a prior thorough training or fine tuning period. Furthermore, most of the alternative techniques have to be trained with completely clean datasets. This requires the audit data traffic to be appropriately preprocessed and cleaned in order to yield meaningful results [19].

In this work, we propose a novel BPA methodology able to automatically adapt its detection capabilities to the current characteristics of the wireless network, without intervention from an IDS administrator. We have developed a multi-layer based application, written in the C language, able to classify network frames as normal or malicious, with very high accuracy. The proposed method only requires a lightweight process of generating a baseline profile of normal utilisation, in order to generate high intrusion detection accuracy and low number of false alarms. The processing requirements of the proposed methodology allow for implementing the detection in real-time.

The aim of our methodology requires the system to be computationally low cost, scalable and applicable to other wireless technologies. The methodology has been tested with two different types of attack, an Opportunistic Injection at the physical layer and a Deauthentication attack, both requiring the prior spoofing of a legal user identity.

The paper is organised as follows. In section II, the most relevant work is reviewed. A brief description of the D-S is presented in section III. In section IV, the proposed algorithms

This work has been funded by the Engineering and Physical Sciences Research Council (EPSRC).

for belief assignment and the concepts about the sliding window approach are explained. The methodology, testbed, and attack scenarios are presented in section V. In section VI, the obtained results are discussed. Finally, conclusions are given in section VII.

II. Related Work

The application of D-S theory for improving the performance of IDSs is a very active research topic. One of the most thorough descriptions of D-S is presented in [12]. The authors present a comparative study between D-S theory and Bayesian inference as data fusion algorithms.

Among all the works on IDS that investigate the use of D-S theory, there exist multiple ways of assigning probabilities to each of the hypotheses. For instance, [18] utilises expert opinion to manually assign the belief probabilities. This BPA process is completely subjective and might not be adequate for automatic and self-adaptive IDSs.

In [4], the authors describe a cross-layer methodology to increase the performance of an anomaly IDS, detecting DoS attacks, based on a real wireless mesh networks. The system uses three different machine learning algorithms, all of which require prior thorough training to be efficient. These are Bayesian network, Decision tree, and Support Vector Machine. Similar to our work, they use a sliding window scheme and compare the efficiency of a cross-layer approach against a single layer approach. However, the data fusion is not carried out using D-S theory.

In [5], the authors present a prototype for Distributed DoS detection over wired link, based on D-S theory. The system, periodically, fuses the knowledge collected from different sensors within the network, in order to infer the current state of the monitored network. The authors express the BPA as three fixed functions. Again, the BPA process is based on previous experiments and subjectivity of the IDS administrator. As the three functions have a fixed shape, this method might be inadequate for automatic and self-adaptive IDSs.

The authors in [7] present and evaluate an IDS for detecting DoS attacks by seeking changes in the Signal-to-Noise Ratio. The value of this single metric is measured from distinct nodes running two different local algorithms, Single Threshold and Cumulative sum. Based on the measured information, their system generates the BPAs through the use of a linear function. The BPAs are fused with D-S theory. The experiments in this work were also carried out in a real IEEE 802.11 testbed.

Another example, [13] proposes two different ways of assigning belief probabilities, for two different datasets. In the first case, their method calculates a threshold based on the length of the dataset, and then utilises that threshold and fixed functions to assign the belief probabilities. In the second case, a scaled approach with pre-defined beliefs is used. In contrast to our work, the mechanisms proposed in [13] would be unable to automatically adjust to changes in the dataset profile, without the intervention of the IDS administrator, because the use of fixed functions and pre-defined threshold.

The methodology employed by [8] uses data mining techniques to proceed with the BPA tasks. The use of data mining techniques mostly focuses on processing large amounts of audit data traffic rather than performing real-time detection. Apart from that, the audit data traffic must be appropriately preprocessed and cleaned in order for a data mining techniques to yield prominent results in intrusion detection [19].

From the presented results, all of these methods are effective in increasing the detection rate and reducing the number of false alarms of the IDSs. However, none of the referred works investigate methods to find an automatic and self-adaptive process of BPA, and few of them could be used off-the-shelf without a previous training or fine tuning period.

On one hand, systems that make use of data mining techniques for BPA require the gathering of large amounts of data traffic, processing it and complete a training period before being able to perform intrusion detection tasks. These systems are unable to automatically adapt to changes in the network traffic behaviour in real-time. On the other hand, systems that have been empirically assigned fixed probability values by the IDS administrator, or systems that employ fixed functions to assign the belief probabilities are unable to automatically adjust to changes in the network traffic behaviour, without the intervention of the IDS administrator.

In this work, we propose a network-based anomaly detection system, which uses a novel statistical-based BPA assignment scheme methodology able to automatically adapt its detection capabilities to the current characteristics of the different metrics of the wireless network, without intervention for an IDS administrator.

III. Dempster-Shafer Theory

The D-S theory of evidence is a mathematical discipline that combines evidence of information from multiple and heterogeneous events in order to calculate the belief of occurrence of another event. We have presented a more detailed description of D-S theory in our previous work [11], along with a comprehensible practical example of D-S.

D-S starts by assuming a Frame of Discernment $\Theta = \{\theta_1, \theta_2, ..., \theta_n\}$, the finite set of all possible mutually exclusive propositions about some problem domain. With regards to this work, the frame of discernment is comprised of $A = Attack$ and $N = Normal$. Assuming Θ has two outcomes $\{A, N\}$, the total number of subsets of Θ, defined by the number of hypotheses that it composes, is $2^{\Theta} = \{A, N, \{A|N\}, \emptyset\}$. In which $\{A|N\} = Uncertainty$.

Each hypothesis from Θ receives from an observer a probability or belief within [0, 1]. This is known as the Basic Probability Assignment (BPA). The function $m(A)$ is defined as A's basic probability number. It describes the measure of belief that is committed exactly to hypothesis A.

$$m: 2^{\Theta} \rightarrow [0,1] \qquad if \begin{cases} m(\emptyset) = 0 \\ m(A) \geq 0, \forall A \subseteq \Theta \\ \sum_{A \subseteq \Theta} m(A) = 1 \end{cases} \tag{1}$$

Let m_1 and m_2 be the BPAs from observer 1 and 2 respectively. Their orthogonal $sum, m = m_1 \oplus m_2$, is called Dempster's rule of combination, and is defined as

$$m(A) = \frac{\sum_{X \cap Y = A} m_1(X) * m_2(Y)}{1 - \sum_{X \cap Y = \emptyset} m_1(X) * m_2(Y)} \quad \forall A \neq \emptyset \tag{2}$$

Among different data fusion methods, the D-S theory of evidence has been chosen in this work for three main reasons. Firstly, D-S is able to combine evidence from multiple and heterogeneous sources. Second, D-S is suitable for detecting previously unseen attacks because it does not require a priori knowledge. Finally, and more importantly, D-S provides the ability of managing and assigning probability to *Uncertainty*, which allows a large range of problems to be tacked.

The authors in [5, 12] present a comparative study of different data fusion methods. This work concludes that D-S theory is more promising than Bayesian inference. The Bayesian approach provides a powerful method to provide final conclusions. However, this method requires complete knowledge of the conditional probabilities of all of the used metrics and specification of the a priori probability distribution of the different hypotheses. This last requirement is unfeasible in many applications [18]. Additionally, the Bayesian method does not allow allocation of probability to *Uncertainty* but only to the hypotheses *Normal* or *Attack* [13].

IV. Belief Generator Mechanisms

In order for the proposed methodology to be efficient, two conditions must be assured. First, the number of legal frames must be larger than the malicious frames. Normal data is more predominant than malicious data in real network traffic [14]. Second, the difference between metrics of legal and malicious frames must be statistically differentiable and quantifiable. Regarding the first assumption, we have tested scenarios in which the proposed system performs with high detection rate even if there are malicious frames in the initial profiling set.

A. Sliding Window Scheme

Before being able to identify any attack, the IDSs need to define what is considered as normal traffic. The proposed system operates on a sliding window scheme using incoming frames from the legal client.

The content of the n frames within the sliding window composes the profiling dataset of the system, every time a new frame is analysed. For each new incoming frame, different metrics are extracted and different statistical parameters are calculated from the metrics. These statistical parameters are used as a reference of normality for assigning the beliefs by the method proposed below. The system has one sliding window for each used metric.

In order to avoid malicious frames altering the reference of normality, the system slides the window only if the current analysed frame has been classified as legal. Otherwise, the sliding window stays static, drops the frame classified as malicious and replaces the last slot in the sliding window with the next incoming frame. Fig.1 represents an example of a sliding window with $n = 20$ frames, in which the 20^{th} frame has been classified as malicious and the sliding window stays static, only replacing the malicious frame.

Nonetheless, malicious frames could still alter the reference of normality. The system needs to capture n frames prior being able to carry out the detection of new incoming frames. The very first sliding window could contain malicious frames. In the case where the majority of frames in the initial window are malicious, the detection mechanism would misclassify. This fact will influence the overall detection performance of the system. As explained in Section VI, the proposed methodology produces good results, even if there exist malicious frames within the first sliding window.

Because the proposed system operates on a sliding window scheme, finding the optimum sliding window length is very important. The length n of the sliding window will influence the overall detection performance of the system. A long window would include, on average, a larger proportion of legal frames and the statistics would average out to represent the normal profile. However, a larger the window length will also slow down the detection process. In contrast, there exists direct correlation between the lightness of the profiling dataset and the chances of misclassification. In our experiments, we have used $n = 31$ as the length n of the sliding window. As demonstrated in Section VI, the proposed methodology generates meaningful results using the sliding window length $31 < n$.

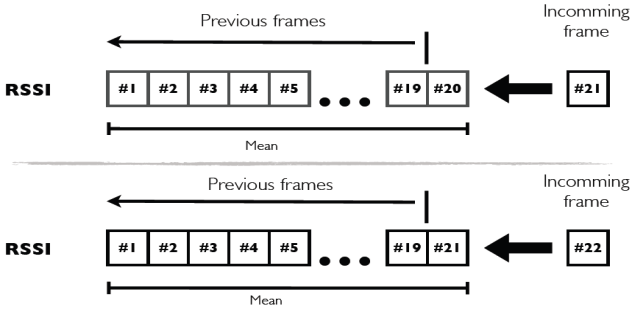

Figure 1. Sliding window scheme.

B. Method to Assign Belief in Attack

We propose three different methodologies for assigning the belief to each hypothesis of Θ, $2^{\Theta} = \{Attack, Normal, Uncertainty, \emptyset\}$. One method generates the belief in *Attack*, and a second method generates the belief in *Normal*. Both work concurrently. Then, based on the belief in *Normal* and *Attack*, a third method calculates a balanced belief in *Uncertainty*.

The methodology that we propose assigns beliefs in *Attack* based on two factors, the Euclidean distance of the current frame from the mean, and frequency of the data. The system calculates an angle α with the distance and the frequency, in order to correlate both factors.

Let us consider a dataset of length n. The system calculates the mean and the highest number of times that a metric is repeated, known as the frequency (F), for the n elements in the dataset. Then, the system calculates the angle α generated by the frequency and the value with the largest Euclidean distance (D_{max}) from the mean, as represented in Fig.2. This angle α represents the maximum possible belief in *Attack*.

For each new incoming frame, the system calculates the angle β generated by F and the distance (D) of this value from the mean. The angle β would be bounded by 0 and α, $0 \leq \beta \leq \alpha$.

Due to the way D-S theory and the BPA are assigned in our methodology, the maximum possible belief in both, *Normal* and *Attack*, is set to 50%.

$$\beta = \cos^{-1}\left(\frac{F}{(D^2 + F^2)^{\frac{1}{2}}}\right) \quad (2)$$

Using a simple linear function, the system then assigns the belief in *Attack*. The minimum belief in *Attack*, 0%, is defined by the angle 0 radians.

C. Method to Assign Belief in Normal

The methodology that we propose assigns beliefs in *Normal*, based on the degree of dispersion of the values in the dataset. The system makes use of quartiles to create classes within the dataset and assigns a fixed belief to each of class.

Let us consider a dataset of length n, sorted in an ascending way. From this sorted dataset, the first quartile (Q_1) will define the boundary for the lower 25% of the data, the second quartile, or median (Me), will define the boundary for the 50% of the data, and the third quartile (Q_3) will define the boundary for the lower 75% of the data.

Similar to the process used with the 'box and whisker' method [15], the Min and Max values are respectively calculated using the following equations:

$$Min = Q_1 - 1{,}5 \times IQR \tag{3}$$

$$Max = Q_3 + 1{,}5 \times IQR \tag{4}$$

where the Interquartile Range (IQR) is the difference between Q_3 and Q_1.

$$\text{IQR} = Q_3 - Q_1 \tag{5}$$

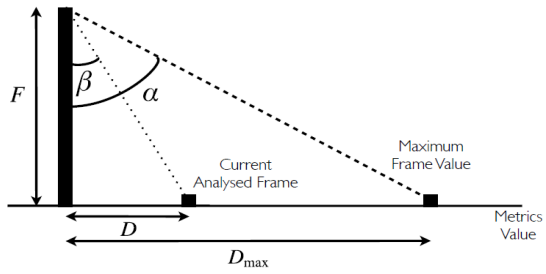

Figure 2. BPA method for belief in Attack.

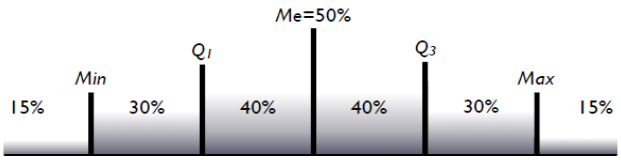

Figure 3. BPA scale for belief in Normal.

Each of the values defines the boundaries of different classes. The metrics of each new incoming frame are allocated within one of these classes. Depending on the class that the current frame is allocated to, the system assigns the belief in *Normal*.

Fig.3 illustrates the different classes and the belief value associated to each of them. If the value of current frame coincides with Me, the belief is 50%. If the value is allocated between the Q_1 and Me, or Q_3 and Me, the belief in *Normal* is 40%. Values between Min and Q_1, or Q_3 and Max will acquire belief of 30%. The rest of the values will acquire belief of 15% in *Normal*.

D. Method to Assign Belief in Uncertainty

Based on the outcome of the two previous methods, the proposed methodology assigns beliefs in *Uncertainty*. The *Uncertainty* is considered in this work as an adjustment parameter.

The outcome of the two previous methods could provide four different conclusions:

- Low belief in *Attack* and high belief in *Normal*.
- High belief in *Attack* and low belief in *Normal*.
- High belief in *Attack* and high belief in *Normal*.
- Low belief in *Attack* and low belief in *Normal*.

For the first and second cases, both methods have reached consistent conclusions. Hence, it is expected that the belief in *Uncertainty* must be low. In contrast, in the third and fourth cases, both methods have reached contradictory conclusions. Therefore, the expected belief in *Uncertainty* is expected to be high in both cases.

We propose the following method for assigning the belief in *Uncertainty*. First, a provisional value is assigned to *Uncertainty* using a linear correlation between the belief in *Normal* and *Attack*. This is:

$$Belief_{Unc.} = \frac{0.5 * Belief_{Min}}{Belief_{Max}} \tag{6}$$

where $Belief_{Max}$ and $Belief_{Min}$ are the larger and the lower of both beliefs, *Normal* and *Attack*, respectively.

As mentioned above, the maximum possible belief corresponds to 0.5. Therefore, belief in *Uncertainty* is also scaled to a maximum of 50%. For instance, if the belief in *Normal* and *Attack* are 0.4 and 0.497, respectively, the value for *Uncertainty* would be: $Belief_{Unc.} = 0.5 * 0.4/0.497 = 0.402$.

Using this example, the summation of all the beliefs is higher than 1. This breaks one of the conditions in the definition of BPA by the D-S theory, explained in (1).

Therefore, an adjustment value μ is calculated as follows:

$$\mu = \frac{X - 1}{3} \tag{7}$$

where X is the summation of the three beliefs. Continuing with the previous example, $X = 0.4 + 0.497 + 0.402 = 1.22$. Then, the adjustment value is $\mu = (1.229 - 1)/3 = 0.099$. Therefore, the beliefs in *Normal*, *Attack* and *Uncertainty* are readjusted to 0.3, 0.397 and 0.303, respectively.

V. System Methodology

For the purpose of this work, we have tested our proposed approach in an IEEE 802.11 network composed of four different parties. These are an Access Point (AP); a monitoring node utilising the TShark [17] software for collecting frames; an attacker; and a client associated with the AP, accessing various websites hosted on the Internet across different geographical locations. The fact that the attacker was placed very close to the AP, around 1.5 meters away, may degrade the detection accuracy. The monitoring node is responsible for performing the proposed intrusion detection. When the monitoring node captures any frame destined to the clients, TShark identifies and isolates the respective metrics of each frame. Then, the gathered dataset is passed to the data fusion process to be analysed. The monitoring node and the attacker were running

Linux and all the devices except from the Linksys WRT54GL AP used the Atheros chipset in their wireless cards.

A. Metrics

We have generated our own IEEE 802.11 dataset for the development and assessment of this work. The captured frames are stored in *pcap* format. Among all the available metrics, four have been experimentally selected as the most appropriate for detecting the attacks. These are the Received Signal Strength Indication (RSSI) at the PHY layer, the Injection Rate and the Network Allocation Vector (NAV) value at the MAC layer, and the Time To Live (TTL) value at the Network layer.

Due to the fact that management and control frames include only information from the PHY and MAC layers in which TTL is not included, in the deauthentication experiments TTL has been replaced by the Delta Time (ΔTime) value. The ΔTime is defined as the time difference between two consecutive frames.

B. Attacks Description

1) Opportunistic Injection Attack

The presented methodology has been tested with two different types of opportunistic injection attacks at the PHY layer. Both injection attacks were implemented with the Airpwn tool. Firstly, Airpwn spoofs the AP MAC address. Then, it listens for legal requests from the clients and injects its own crafted HTML code. In the first type (Attack01), the attacker replaces the whole content of the website. In the second type (Attack02) the attacker replaces only the images in the website. Both could cause harm of varying severity, i.e. to redirect the client to a phishing website.

2) Deauthentication Attack

Another type of attack that has been investigated is the deauthentication of wireless clients from the legal AP. This type of attack is commonly utilised in DoS attacks but also constitutes the first step of breaking into WPA2 encrypted wireless networks. In the latter case, the attacker injects a few spoofed deauthentication frames with the purpose of forcing the client to re-establish a connection with the AP. At a later stage and off-line, the attacker could succeed in cracking WPA2 by applying brute force or dictionary attack techniques. The suite of tools used to implement this attack is Aircrack.

The detection of the deauthentication attack was possible just by using management and control frames for two reasons. Firstly, deauthentication attacks are highly correlated with information in the management frames. Furthermore, because the network was encrypted with WPA2, and with the assumption that the monitor node does not have the key, it was not possible or necessary to retrieve information above the MAC layer.

C. Evaluation Mechanisms

In order to evaluate the effectiveness of the proposed methodology, the results from the multi-layer scheme are compared against the same methodology, but utilising fewer metrics. All cases have been evaluated with the same gathered dataset.

The results are evaluated by comparing the False Negative Rate (FN_{Rate}), False Positive Rate (FP_{Rate}) and Detection Rate (DR). These are:

$$DR = TP/(TP + FN) \quad (8)$$

$$FN_{Rate} = FN/(TP + FN) \quad (9)$$

$$FP_{Rate} = FP/(TP + FP + TN + FN) \quad (10)$$

where True Positive (TP) is the number of attack frames correctly classified as malicious, True Negative (TN) is the number of non-attack frames correctly classified as legal frames, False Positive (FP) is the number of non-attack frames misclassified as malicious, and False Negative (FN) is the number of attack frames misclassified as legal frames.

VI. RESULTS

The experimental results are presented in the form of 'Bar Charts'. The Y-axis of the graphs represents the percentage of DR and FP. The X-axis of the graphs represents the index of the used metrics. Each index corresponds to one possible combination of metrics, with #1 being the set that combines all the considered metrics and set #15 a single metric set. Therefore, the best results are to be expected from the test index #1.

The indexes of all the possible sets metric combinations are presented in Table I.

TABLE I. INDEXES OF THE USED METRICS

Index-Metrics	Index-Metrics	Index-Metrics
#1-RSSI-Inj.Rate-TTL-NAV[a]	**#6**-Inj.Rate-TTL[a]	**#11**-Rate-NAV
#2-RSSI-Inj.Rate-TTL[a]	**#7**-TTL-NAV[a]	**#12**-RSSI
#3-RSSI-Inj.Rate-NAV	**#8**-RSSI-TTL[a]	**#13**-Inj.Rate
#4-Inj.Rate-TTL-NAV[a]	**#9**-RSSI-Inj.Rate	**#14**-TTL[a]
#5-RSSI-TTL-NAV[a]	**#10**-RSSI-NAV	**#15**-NAV

a. TTL is replaced by ΔTime in Deauthentication Attack Results.

A. Opportunistic Injection Attack Results

The multi-layer results for the injection attack experiments with non-attack traffic, using all the metric combinations, are presented in Fig.4. As expected, none of the combinations produce attack detection. DR is 0% in all the cases. In terms of FPs, the detection results using all the considered metrics (#1) are the best results, generating 0% of FP. Using fewer numbers of metrics produces higher numbers of FPs. As can be seen, the combination of RSSI-Inj.Rate-TTL (#2) and RSSI-TTL-NAV (#5) generate 1.21% and 1.18%, respectively. Furthermore, the use of a dual metric, RSSI-TTL (#8), and single metric, RSSI (#12) generates the most significant results, with 33.7% and 35.1% of FPs, respectively.

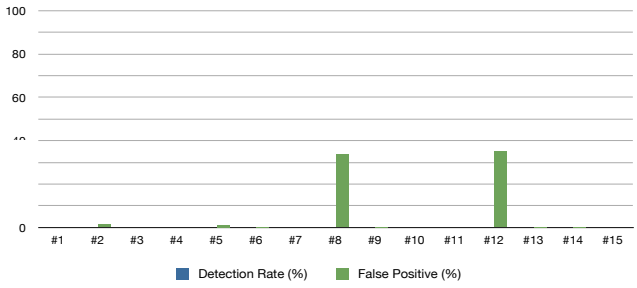

Figure 4. Injection Attack Results with Non-Attack Traffic.

The results for the Attack01 experiments, using all the metrics combinations, are presented in Fig.5. In these experiments, all the metric combinations produce 100% DR. Similar to the

previous results, the set of metrics RSSI-Inj.Rate-TTL (#2), RSSI-TTL-NAV (#5), RSSI-TTL (#8) and RSSI (#12) are the only sets that produce FPs. In this set of experiments, the combination of three metric, #2 and #5, both produce 0.37% of FPs. More significant, set #8 generates 39.13% of FPs, and the single metric #12 generates 69.28% of FPs. Also in these experiments, the multi-layer results are the best overall.

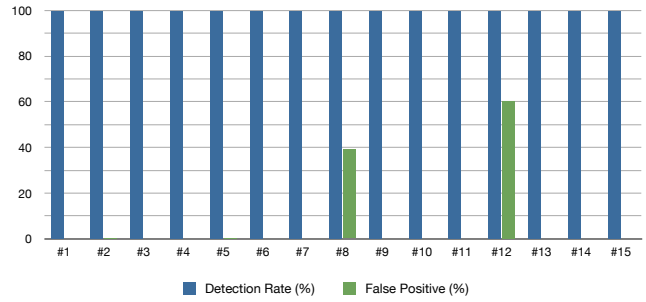

Figure 5. Injection Attack Results with Non-Attack Traffic.

Fig.6 and Fig.7 illustrate the results for the Attack02 experiments and the results when both types of injection attacks are launched together, respectively. Similar to the results for the Attack01 experiments, the set of metrics RSSI-TTL (#8) and RSSI (#12) are the sets with higher numbers of FPs. Set #8 produces 25.29% of FPs and the single metric #12 exceeds 41% of FPs, in the Attack02 experiments. None of the other possible combinations reaches 0.5% of FPs. In terms of DR, only the results of single metric #12 are lower than 100%. Again, the multi-layer results are the best results overall.

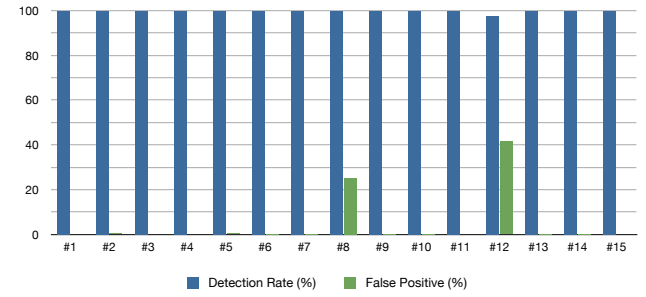

Figure 6. Injection Attack Results with Attack 02 Traffic.

In the experiments with both attacks, the set #8 exceeds 39.9% of FPs and the single metric #12 exceeds 53.4% of FPs. None of the other possible combinations reaches 0.1% of FPs, but the two cases #2 and #5 that exceed 2% of FPs. In terms of DR, all the combinations produce perfect results. As expected, the multi-layer results are the best results.

From the presented results of single metrics, Inj. Rate (#13), TTL (#14) and NAV (#15), it is doubtful that there is any benefit utilising a combination of different metrics. Each of the three cases produces perfect detection, as does the multi-layer combination case. Additionally, all the cases that produce high rates of false alarms make use of the metric RSSI. This fact indicates that the utilisation of the metric RSSI tends to deteriorate the results. However, as presented below for the case of the deauthentication attack, the use of the metric RSSI helps to improve the results for the multi-layer approach, and the results using all the considered metrics are the best results overall.

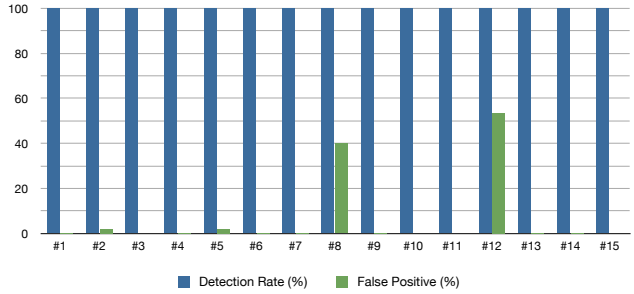

Figure 7. Injection Attack Results with Attack 01-Attack 02 Traffic.

B. *Deauthentication Attack Results*

As explained in Section V, the communication between the client and the AP was protected with WPA2 and a pre-shared key. Even though the system utilises metrics from just the two lower layers of the protocol stack, the presented methodology was able to detect the deauthentication attack.

The multi-layer results for the deauthentication attack experiments, using the four considered metrics, are presented in Fig.8. As can be appreciated, these are the best results overall in detecting the attacks. The detection system generates 0% of FNs and 2.17% of FPs. Similar to the experiments with injection attack, the FPs are caused because the analysed metrics of the malicious frames are very close to the legal frames.

In contrast to the experiments with injection attacks, the use of RSSI benefits the results because the number of FPs have been reduced. The metrics Inj.Rate and ΔTime have also a different influence over the generated results. In these experiments, the ΔTime increases the number of FPs, and the Inj.Rate is completely ineffective in detecting the deauthentication attack. This is because management frame, both legal and malicious, are transmitted at a fixed rate of 1 Mbps. As can be seen in Fig.8, the use of Inj.Rate degrades the DR results of all the possible metric combinations; specially, the sets Inj.Rate-ΔTime (#6), RSSI-Inj.Rate (#9), Inj.Rate-NAV (#11) and Inj.Rate (#13). However, when using the four metric multi-layer approach, the use of Inj.Rate preservers high DR and does not degrade the results in terms of FP.

In addition, the use of Inj.Rate benefits the results because it reduces the number of FPs. In particular, the use of RSSI-ΔTime-NAV produces 0% of FNs but 9.13% of FPs. When including the Inj.Rate, the number of FPs is reduced to 2.17%.

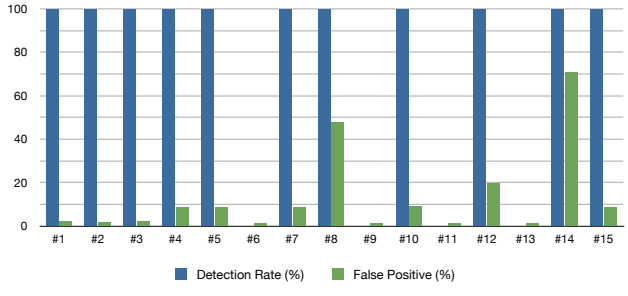

Figure 8. Deauthentication Attack Results.

C. Optimum Sliding Window Length

In order to find the optimum sliding window length, the detection process has been carried out for all the possible lengths between 1 and 150 frames. For each of the iteration, the length of the sliding window was increased by one frame. The analysis has been performed in an off-line manner, using the same datasets in all the iterations. The selection of the optimum sliding window length is based on four parameters, the sliding window length, the percentage of DR and FP results, and the processing time required to obtain the detection results.

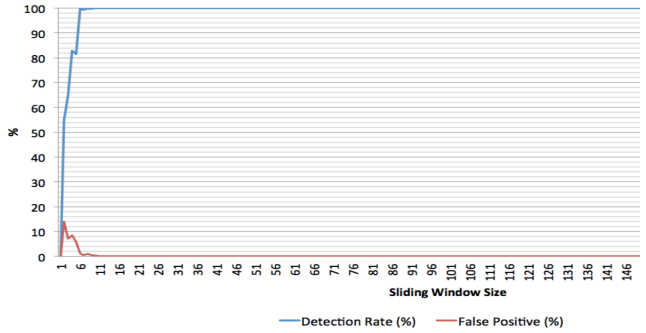

Figure 9. Multi-Layer Results Injection Attack02 Variable Window Size.

The multi-layer results for the Attack02 experiments are shown in Fig.9. If the length of the sliding window is $n < 12$, the detection performance is highly degraded. On the other hand, any length n above 12 frames, $n \geq 12$, produces perfect detection, with 100% of DR and 0% of FP. The processing time linearly increases with the value of n. If $n = 1$, the system requires 3µsec to obtain a decision whether the analysed frame is malicious or not. If $n = 150$, the same system requires 173µsec to obtain a final decision. Given that the average interarrival time is 2.37msec, any sliding window length could produce results in real-time.

The results for the rest of the injection attack experiments are not shown. The performance of these experiments was similar to the results just presented, sharing $n = 12$ as the minimum sliding window length for generating perfect detection in all the injection attack experiments.

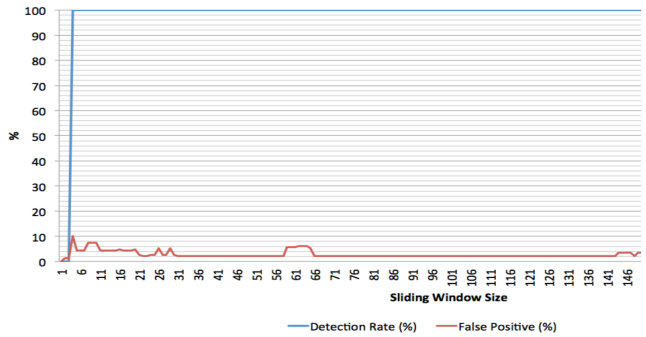

Figure 10. Multi-Layer Results Deauthentication Variable Window Size.

A similar approach has been implemented with the deauthentication attack. The multi-layer results for the deauthentication attack experiments are shown in Fig.10. If the length of the sliding window is $n < 31$, the results are highly inconsistent, in terms of FPs. The system produces almost perfect detection, when the length n value is higher than 31 frames, $n \geq 31$, with 100% of DR and 2.17% of FP. In terms of processing time, if $n = 1$, the system requires 6µsec to obtain a decision for these experiments. Meanwhile, if $n = 150$, the system consumes 445µsec in obtaining a final decision. The results could also be calculated in real-time.

In our experiments, we have used $n = 31$ as the length n of the sliding window, because it is the minimum value that produces the best results in all the experiments. However, any value above 31 could be chosen.

D. Malicious Frames within the Initial Sliding Window

Apart from the prior thorough training or fine tuning period, other proposed algorithms, such as data mining techniques, require training with completely clean datasets. Generally, the term clean dataset refers to two different meanings, handling missing data and removing spurious data [19]. This work has only considered the effect that spurious frames have over the detection performance.

In order to generate meaningful results, the detection systems based on data mining techniques require a training period using clean datasets. Otherwise, the detection results would be highly inaccurate. Our proposed methodology produces good results, even if there exist malicious frames within the first sliding window. Also, the proposed methodology drops the frames classified as malicious, to avoid malicious frames altering the reference of normality.

The percentage of malicious frames within the first sliding window influences massively the detection results of the proposed methodology. In order to calculate the maximum number of malicious frames accepted by our methodology, the detection process has been repeated several times, introducing one additional new malicious frame for each of the iteration. The results of these experiments are based on the percentage of DR results, the percentage of malicious frames within the initial sliding window, and the sliding window length.

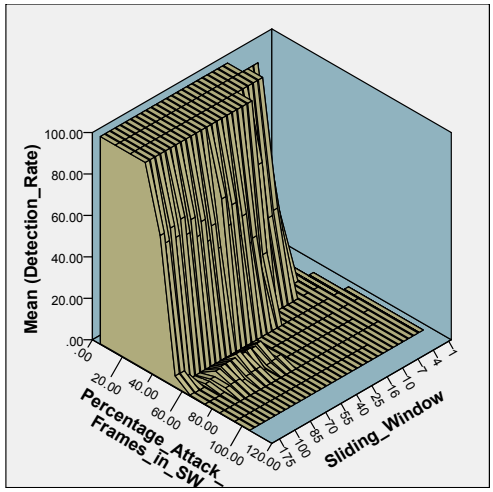

Figure 11. Percentage of Malicious Frames in the First SW - Attack01.

The multi-layer results for the Attack01, Attack02 and the experiments when both types of injection attacks are launched together, are shown in Fig.11, Fig.12, and Fig.13, respectively. There exists an evident consistency in the results of all these

experiments. For any sliding window length larger than 12 frames, $n \geq 12$, the detection system produces perfect detection with up to 43% of malicious frames within the initial sliding window. A higher percentage of malicious frames makes the detection accuracy to drastically drop.

Fig.14 shows the multi-layer results for the deauthentication attacks. The detection system produces perfect detection with up to 20% of malicious frames within the initial sliding window, if the length of the sliding window is between 31 and 90 frames, $31 \leq n \leq 90$. If the length of the sliding window is $n < 90$, the system produces perfect detection with up to 13% of malicious frames within the initial sliding window. Remarkable is the fact that, for any amount of malicious frames within the initial sliding window and for any sliding window length smaller than 31 frames, $n < 31$, the detection system is unable to produce higher DR than 95%. In contrast to the injection attacks experiments, the detection accuracy gradually drops along with the percentage of malicious frames within the initial sliding window.

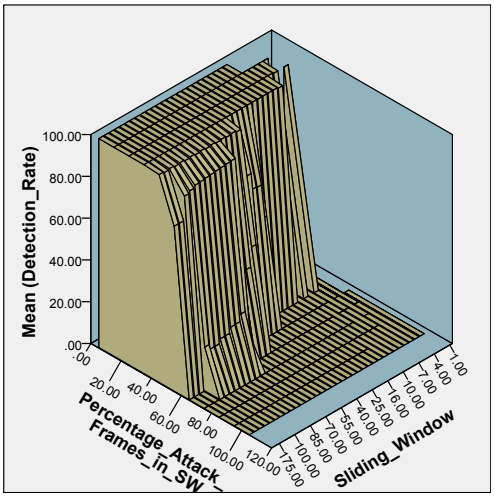

Figure 12. Percentage of Malicious Frames in the First SW - Attack02.

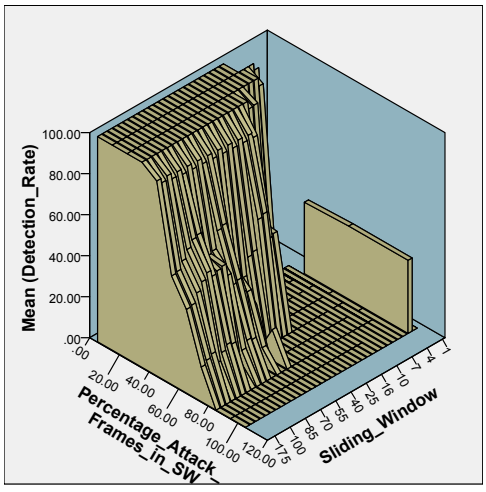

Figure 13. Percentage of Malicious Frames in the First SW - Attack01-Attack02.

VII. CONCLUSIONS

This paper tackles an important step that remains open in the use of D-S theory in network security infrastructures, finding an automatic and self-adaptive process of Basic Probability Assignment. The authors of this work have proposed and evaluated a novel BPA methodology able to automatically adapt its detection capabilities to the current characteristics of the wireless network, without intervention from an IDS administrator for selecting thresholds or manually/experimentally assigning beliefs. The system only requires a light profiling process of 31 frames. An analysis has been carried out a per frame bases.

The proposed methodology has been evaluated with real WiFi data traffic in a testbed environment. Two different types of attack have been investigated; an Opportunistic Injection attack and a Deauthentication attack. In order to evaluate the effectiveness of the proposed methodology, the results from the multi-layer methodology are compared against the same methodology, but utilising metrics from fewer layers.

As explained throughout this work, using the proposed methodology, there exist some injection attacks that can be easily detected by utilising the information from only one single metric. However, using solely the same single metric for detecting a deauthentication attack would be highly ineffective. In both attack scenarios, the combination of information from all the metrics produces the best results overall in detecting malicious injected frames. By considering these results, it is clear that the proposed manner of intelligent combination of beliefs from different metrics yields an improved performance, in terms of detection rate and false alarms

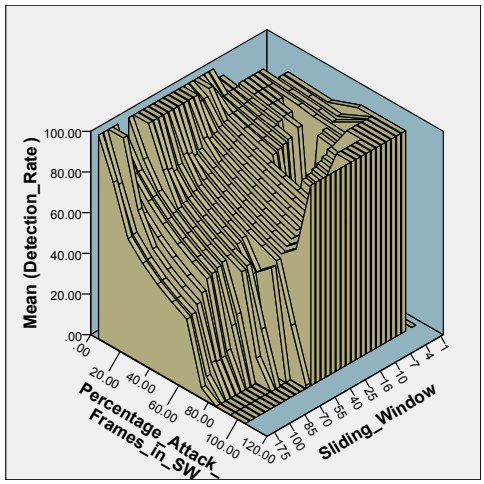

Figure 14. Percentage of Malicious Frames in the First SW - Deauth Attack.

The classification of a single frame takes from 3μsec to a couple of hundreds μsec, allowing the detection to be carried out in real-time. As has been demonstrated in this work, our proposed methodology is able to produce perfect detection even if the dataset used as a reference of normality contains malicious frames.

As for future work, we will investigate methods to automatically select the most appropriate metrics for detecting each of the attacks taking place. This will help to reduce the number of false alarms and towards better mitigation techniques. Finally,

the proposed technique can be installed in multiple monitors that could collaboratively work for achieving higher DR.

REFERENCES

[1] K. Zeng, K. Govindan, D. Wu, and P. Mohapatra, "Identity-based attack detection in mobile wireless networks," in *Proc. INFOCOM 2011*, Shanghai, China. April 10-15. 2011, pp. 372–377.

[2] Q. Li, and W. Trappe, "Relationship-based detection of spoofing-related anomalous traffic in ad hoc networks," in *Proc. SECON 2006*, Reston, VA, USA. September 28. 2006, pp. 50–59.

[3] G. Thamilarasu, S. Mishra, and R. Sridhar, "A cross-layer approach to detect jamming attacks in wireless ad hoc networks," in *Proc. MILCOM 2006*, Washington, DC, USA. October 23-27. 2006, pp. 1–7.

[4] X. Wang, J. S. Wong, F. Stanley, and S. Basu, "Cross-layer based anomaly detection in wireless mesh networks," in *Proc. SAINT 2009*, Bellevue, Washington, USA. July 20-24. 2009, pp. 9–15.

[5] C. Siaterlis, and B. Maglaris, "Towards multisensor data fusion for DoS detection," in *Proc. SAC 2004*, Nicosia, Cyprus. March 14-17. 2004.

[6] V. Chatzigiannakis, G. Androulidakis, K. Pelechrrinis, S. Papavassiliou, and V. Maglaris, "Data fusion algorithms for network anomaly detection: classification and evaluation," in *Proc. ICNS 2007*, Athens, Greece. June 19-25. 2007, pp. 50–56.

[7] A. G. Fragkiadakis, V. A. Siris, and A. P. Traganitis, "Effective and robust detection of jamming attacks," in Proc. *Future Network and MobileSummit 2010*, Florence, Italy. June 16-18. 2010, pp. 1–8.

[8] D. Yu, and D. Frincke, "Alert confidence fusion in intrusion detection systems with extended dempster-shafer theory," in *Proc. ACM 2005*, Kennesaw, GA, USA. March 18-20. 2005, pp. 142–147.

[9] Y. Sheng, K. Tan, G. Chen, D. Kotz, and A. Campbell, "Detecting 802.11 MAC layer spoofing using received signal strength," in *Proc. INFOCOM 2008*, Phoenix, USA. April 13-18. 2008, pp. 1768–1776.

[10] G. Chandrasekaran, J.-A. Francisco, V. Ganapathy, M. Gruteser, and W. Trappe, "Detecting identity spoofs in IEEE 802.11e wireless networks," in *Proc. GLOBECOM 2009*, Honolulu, USA. Nov 30-Dec 4. 2009.

[11] F. J. Aparicio-Navarro, K. G. Kyriakopoulos, and D. J. Parish, "An on-line wireless attack detection system using multi-layer data fusion," in *Proc. M&N 2011*, Anacapri, Italy. October 10-11. 2011, pp. 1–5.

[12] T. M. Chen, and V. Venkataramanan, "Dempster-Shafer theory for intrusion detection in ad hoc networks," *IEEE Internet Computing*, vol. 9, no. 6, pp. 35–41, Nov-Dec. 2005.

[13] Q. Chen, and U. Aickelin, "Anomaly detection using the dempster-shafer method," in *Proc. DMIN 2006*, Las Vegas, Nevada, USA. June 26-29. 2006, pp. 232–240.

[14] C. Thomas, and N. Balakrishnan, "Improvement in minority attack detection with skewness in network traffic," in *Proc. SPIE 2008*, Orlando, Florida, USA. March 17. 2008.

[15] C.-C. Tuan, Y.-C. Wu, and W.-S. Chang, "Fault tolerance by quartile method in wireless sensor and actor networks," in *Proc. CISIS 2010*, Krakow, Poland. February 15-18. 2010, pp. 758–763.

[16] C. Thomas, and N. Balakrishnan, "Advanced sensor fusion technique for enhanced intrusion detection," in *Proc. ISI 2008*, Taipei, Taiwan. June 17-20. 2008, pp. 173–178.

[17] G. Combs, "TShark website," http://www.wireshark.org/docs/man-pages/tshark.html (Access Date: 03 Sep, 2012).

[18] J. R. Boston, "A signal detection system based on dempster–shafer theory and comparison to fuzzy detection," *IEEE Systems, Man, and Cybernetics Society*, vol. 30, no. 1, pp. 45–51, 2000.

[19] T. Mitsa, "Temporal data mining," Chapman & Hall/CRC Data Mining and Knowledge Discovery Series, 2010.