Enhancement of Passive MAC Spoofing Detection Techniques

Aiman Abu Samra and Ramzi Abed
Islamic University of Gaza
Gaza, Palestine
{aasamra, rabed}@iugaza.edu.ps

***Abstract*- Failure of addressing all IEEE 802.11i Robust Security Networks (RSNs) vulnerabilities enforces many researchers to revise robust and reliable Wireless Intrusion Detection Techniques (WIDTs).**

In this paper we propose an algorithm to enhance the performance of the correlation of two WIDTs in detecting MAC spoofing Denial of Service (DoS) attacks. The two techniques are the Received Signal Strength Detection Technique (RSSDT) and Round Trip Time Detection Technique (RTTDT). Two sets of experiments were done to evaluate the proposed algorithm. Absence of any false negatives and low number of false positives in all experiments demonstrated the effectiveness of these techniques.

Keywords: Intrusion Detection, RSS, RTT, Denial of Service.

I. INTRODUCTION

Due to the vast interest in WLAN technologies, these wireless networks have matured a lot since ratification of the first 802.11 standard in 1997 [2, 4]. Since then, several amendments have been made to the base standard[1], out of which most have been to the physical (PHY) layer to increase the operating speeds and throughput of WLANs [12]. However, one amendment -IEEE 802.11i was ratified in 2004 to address the threats of confidentiality, integrity and access control in WLANs [16].

As a result of the failure of the WLAN standards to address the lack of authentication of 802.11 Management frames and network card addresses, it is possible for adversaries to spoof the identity of legitimate WLAN nodes and take over their associations. Such attacks, where the attacker assumes the identity of another WLAN node, are referred to as MAC spoofing or simply spoofing based attacks. Such attacks are of grave concern as they can lead to unauthorized access and leakage of sensitive information. MAC spoofing is the root of almost all RSN attacks. Without the ability to inject forged frames using a spoofed MAC address, none of the RSN attacks can be launched. WIDS should be passive, accurate and sensitive [19]. Unfortunately, few intrusion detection techniques are available for reliably and accurately detecting MAC spoofing. The few that exist are not very robust and reliable. Given the enormous impact MAC spoofing has on WLAN security, wireless intrusion detection techniques are required to reliably and accurately detect MAC spoofing activity in WLANs.

II. WIRELESS INTRUSION DETECTION TECHNIQUES FOR MAC SPOOFING

The intrusion detection systems can be divided into two main categories depending on how their events of interest [7, 20, 21]:

Misuse-Based IDSs: Require that patterns representing security events be explicitly defined. This pattern is usually referred to as a signature. The IDS monitors computer systems and networks looking for these signatures and raises an alert when it finds a match.

Anomaly-Based IDSs: Anomaly-based IDSs on the other hand, do not require explicit signatures of security events. They use expected or non-malicious behavior and raise any deviations from this behavior as security events.

RSNs suffer from a number of security vulnerabilities; out of which the ability to spoof a WLAN node's MAC address is the most serious one. MAC spoofing allows an adversary to assume the MAC address of another WLAN node and launch attacks on the WLAN using the identity of the legitimate node. Without this vulnerability, an adversary will not be able to inject forged frames (Management, Control, EAP) into the WLAN and all attacks based on injection of such frames would be impossible [1]. Some of these attacks are Man-in-the-Middle, Session Hijacking, Rogue AP, Security Level Rollback, RSN IE Poisoning, EAP based DoS attacks, Management and Control frame based DoS attacks. Even exploiting the unprotected MAC frame Duration field to cause a DoS (virtual jamming) is also only possible in combination with MAC Spoofing. Software Implementation Based Attacks are also launched when an adversary injects forged frames containing exploit code into the WLAN using the MAC address of another WLAN node. The 4-Way Handshake Blocking and Michael Countermeasures DoS attacks are also launched using forged frames with spoofed MAC addresses.

The use of CCMP for confidentiality and integrity protection in RSNs has removed the threat of eavesdropping based passive attacks such as brute force and other key discovery attacks on the captured WLAN traffic. Hence, most attacks in RSNs are performed using active injection of forged frames into the WLAN using spoofed identity (MAC address)

[1] http://standards.ieee.org/getieee802/802.11.html

of other WLAN nodes. Even attacks that do not use MAC Spoofing directly exploit it in post attack activity. For instance, after an adversary has successfully discovered key material using the Dictionary Attack, it would use MAC Spoofing to authenticate to the WLAN using the key material and the MAC address of the victim node.

Hence, MAC Spoofing is responsible for majority of the attacks on RSNs. Spoofing based attacks in WLANS are possible as the existing WLAN standards fail to address the lack of authentication of unprotected WLAN frames and network card addresses. To further exacerbate the problem, almost all WLAN hardware provides a mechanism to change its MAC address; hence trivializing changing identities.

MAC Spoofing is the root cause of all injection based attacks on RSNs. A number of different techniques have been suggested to detect MAC spoofing activity in a WLAN. These are discussed below:

A. Sequence Number Monitoring:

This approach was first suggested by Wright [102] and was later used by Godber and Dasgupta [10] for detecting rogue APs. Kasarekar and Ramamurthy [13] have suggested using a combination of sequence number checks along with ICMP augmentation for detecting MAC spoofing. The idea is that an adversary spoofing the MAC address of a legitimate node will be assigned the same IP address as the legitimate node by the DHCP server of the WLAN. Hence, an ICMP ping to that IP address will return two replies; clearly identifying existence of MAC spoofing.

Guo and Chiueh [11] extended sequence number based MAC spoofing detection by monitoring patterns of sequence number changes. Rather than raising an alarm if a single sequence number gap is detected for a MAC address, the MAC address is transitioned to a verification mode and the subsequent sequence numbers of that MAC address are monitored for any anomalous gaps. In this manner, false positives raised due to lost and out of order frames are avoided. Their system also caches the last few frames for each MAC address to verify retransmissions and out of order frames.

Their solution also uses regular ARP requests to all STAs to synchronize with their sequence numbers based on ARP responses. This is done to defeat an adversary successfully injecting frames with correct sequence numbers somehow and detect the spoofing even if the legitimate node is no longer transmitting.

Madory [15] suggests a technique called Sequence Number Rate Analysis (SNRA) to detect MAC spoofing using sequence numbers. This technique calculates a transmission rate for a MAC address. If the calculated transmission rate is greater than the theoretical transmission limit for PHY of the WLAN it is considered to be an indication of a MAC spoof.

B. Fingerprinting

Fingerprinting MAC addresses based on their unique characteristics. The combination of device driver, radio chipset and firmware provides each WLAN node a unique fingerprint of its 802.11 implementation. Ellch [5] suggests using CTS frame responses and 802.11 Authentication and Association frames to fingerprint 802.11 implementations of WLAN nodes. He also suggests using the Duration field values in 802.11 frames to fingerprint WLAN nodes in a particular WLAN. Such fingerprints can be used to detect MAC spoofing activity as the fingerprint for the adversary would be different from the legitimate node. Franklin et al. [6] also suggest similar fingerprinting of 802.11 device drivers. Their technique exploits the fact that most 802.11 drivers implement the active scanning algorithm differently. They suggest that each MAC address could be mapped to a single device driver fingerprint and hence could be used for detecting MAC spoofing.

C. Location Determination

Location of the WLAN nodes can also be used to detect MAC spoofing. Location of a particular node is usually determined using its signal strength values as a location dependent metric.

Once the location of a MAC address is known, any changes in its location can be used as an indication of MAC spoofing activity. Bahl and Padmanabhan [4] record the received signal strength (RSS) values of each node on each AP and then compare these against a pre-calculated database that maps these RSS values to physical locations. Smailagic and Kogan [17] improve on this system and use a combination of triangulating WLAN nodes' RSS values from multiple APs and lookups in a database that maps RSS values to physical locations. Many other systems have also been proposed that establish location of a WLAN node using its RSS values and hence can be used for detecting MAC spoofing in a WLAN [4, 6,].

D. Signal Strength Fourier Analysis

Madory [15] also suggests a statistical technique called the Signal Strength Fourier Analysis (SSFA) to detect MAC spoofing using received signal strength (RSS) values of a WLAN node. It performs Discrete Fourier Transform on a sliding window of RSSs and uses the statistical variance of the high-frequencies which result from the interference between the attacker and the victim to detect MAC spoofing.

Some of the techniques for detecting spoofing based attacks have been implemented in some open source WIDSs such as Snort-Wireless [7]. Snort-Wireless claims to be capable of detecting MAC spoofing by monitoring for inconsistencies in MAC frame sequence numbers.

III. RELATED WORK

R. Gill et al. [8] address this issue by proposing two wireless intrusion detection techniques (WIDTs): Received Signal Strength Based Intrusion Detection Technique (RSSDT), and Round Trip Time Based Intrusion Detection Technique (RTTDT). These WIDTs are capable of detecting the spoofing based attacks reliably, and meet many of the desirable characteristics as they: are based on unspoofable characteristics of the PHY and MAC layers of the IEEE 802.11 standard; are passive and do not require modifications to the standard, wireless card drivers, operating system or client software; are computationally inexpensive; and do not interfere with live traffic or network performance.

In [8] RSSDT and RTTDT work effectively against session hijacking attacks where the attacker and the legitimate STA are geographically separated and the differences in observed RSS and RTT between the attacker and the STA are significant. For more reliable intrusion detection, the RSSDT and RTTDT were not used in isolation from each other. Rather results from both the techniques were correlated to provide more confidence in the generated alarms [7].

IV. Experiments

This section demonstrates the accuracy and utility of RSSDT and RTTDT through empirical data and use of correlation techniques. The RSSDT and the RTTDT, both use threshold values, namely the RSSdiff threshold and the RTTdiff threshold respectively. The RTTdiff and RSSdiff values greater than these thresholds are considered anomalous. In these experiments, RSSdiff threshold and the RTTdiff threshold were set to the value of 5.

To assist empirical analysis, eight experiment scenarios per attack were designed to study the effectiveness of the RSSDT and the RTTDT in the presence of an attacker, who launches three different new attacks against a legitimate station (STA); TKIP Cryptographic DoS attack [9, 18], Channel Switch DoS attack [14], and Quite DoS attack [14].

A. *Equipment and Preparation*

The experiments were carried out in a lab environment. The same networking hardware/software was used in all experiment scenarios. The following four parties took part in the scenarios: a legitimate client (STA), an access point (AP), a passive Intrusion Detection System (IDS) sensor, and an attacker.

B. *Correlation Engine*

Results from both the RTTDT and the RSSDT were correlated to provide more confidence in the generated alarms. The correlation engine used was event based i.e. if one of the detection techniques detected an anomaly (an alert), the correlation engine activated and waited until it obtained the detection results from the other technique, before making a decision on whether or not to raise an alarm. If both the detection techniques detected the anomaly, then an alarm will be raised.

C. *Experimentation -Set1*

In *Set1* of the experiments, robustness and reliability of the RTTDT and the RSSDT were tested when none of the participants were in motion. The AP, the IDS sensor, the STA and the attacker were all stationary in this set. In all scenarios, the AP was placed in close proximity to the IDS sensor. In Fig. 1, points A, B and C represent location of the STA, the AP and the IDS sensor respectively. Experimentation set1 has 4 scenarios. Points X,Y and Z represent location of the attacker in Scenarios Two, Three and Four respectively. While scenario four has no attaker.

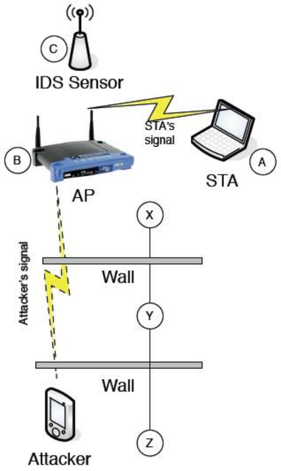

Figure 1. Experimentation Set-1

1) *Scenario One*

In *Scenario One*, there was no attacker present and the AP and the STA were placed in close proximity to each other at points B and A respectively (see Fig. 1). Network traffic was generated from the STA to the AP. The IDS sniffer was used to capture this WLAN traffic between the STA and the AP. After examination of 1000 captured frames, the *correlation engine* did not raise any alarms. As there was no attacker present, both the detection techniques and the correlation engine correctly did not generate any false positives.

2) *Scenario Two*

In *Scenario Two*, the AP and the STA were placed in close proximity to each other at points B and A respectively. The attacker was placed in line of sight of the STA at point X (Fig. 1). Then network traffic was generated between the STA and the AP. The attacker then launched the attack on the STA.

In this scenario three different experiments were carried out; in the first one, the attacker launched a TKIP DoS Attack [9, 18], in the second experiment he launched a Channel Switch DoS attack [14], while in the third experiment the attacker launched a Quite DoS attack [14]. For each experiment, traffic was captured using *Wireshark*[2], after that the captured traffic was examined using the IDS Sensor which is based on the *correlation engine*, which resulted in two alarms.

3) *Scenarios Three and Four*

Similar to scenario two except the location of the attaker it is in point Y for scenario three where three alarms were generated when running. And in point Z for scenario four where tow alarms were generated.

D. *Experimentation -Set2*

In experiments *Set2*, the robustness and reliability of the RTTDT and the RSSDT were tested with the attacker stationary and the STA in motion between a point closer

[2] http://www.wireshark.org

to the AP and another point far away from it. The AP, the IDS sensor, and the attacker were all stationary at locations B, C and D (or G) in Fig. 2. In all scenarios, the IDS sensor was placed in close proximity to the AP.

1) *Scenario Five*

In *Scenario Five*, the AP and the attacker were stationary and were placed in close proximity to each other (in line of sight at points B and D in Fig. 2). Network traffic was then generated from the STA to the AP. The STA then started traveling (at walking pace) from a point close to the AP to a point far away from it (i.e. from point E to F in Fig. 2). Towards the end of the STA's journey, the attacker then launched three different attacks on the STA as described in Scenario Tow. After capturing the traffic; executing IDS Sensor over the captured traffic resulted in two alarms.

2) *Scenario Six*

Similar to scenario Five except the STA waking from point F to E and one alarm was raised.

3) *Scenario Seven*

In *Scenario Seven*, the AP and the attacker were stationary and were placed far away from each other (not in line of sight, at points B and G in Fig. 2). Then similar to scenario Five with one alarm.

4) *Scenario Eight*

Similar to scenario Seven except the STA waking from point F to E. Also one alarm was raised.

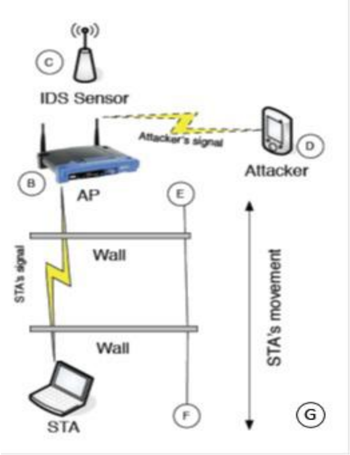

Figure 2. Experimentation Set-2

V. Analysis

True Positives and False Positives

In our experiments, no false negatives were registered. However, some false positives were raised by the correlation engine.

Tables 1, 2, and 3 summarize the true positives raised by the correlation engine when applying TKIP DoS attack, Channel Switch DoS attack, and Quite DoS attack respectively in all eight scenarios. For instance, the entry for Scenario Two in Table 1 shows that when applying the first attack, the true alarm was raised by the correlation engine at frame 499. This alarm was caused by a RSS fluctuation (RSSdiff =19) at frame 499 and a RTT spike at frame 510 (RTTdiff =29.651) for the STA. Frame 510 was the very next RTS-CTS handshake event for the STA after frame 499. Hence, both the RSSDT and the RTTDT sensors reported the anomaly and the TKIP DoS attack was identified correctly and accurately. Also, the entry for Scenario Two in Table 2 shows that when applying the second attack in Scenario Two, the true alarm was raised by the correlation engine at frame 520. This alarm was caused by a RSS fluctuation (RSSdiff =16) at frame 520 and a RTT spike at frame 543 (RTTdiff =25.556) for the STA. Frame 543 was the very next RTS-CTS handshake event for the STA after frame 520. Hence, both the RSSDT and the RTTDT sensors reported the anomaly and the Channel Switch DoS attack was identified correctly and accurately. Moreover, the entry for Scenario Two in Table 3 shows that when applying the third attack, the true alarm was raised by the correlation engine at frame 602. This alarm was caused by a RSS fluctuation (RSSdiff =16) at frame 602 and a RTT spike at frame 620 (RTTdiff =26.447) for the STA. Frame 620 was the very next RTS-CTS handshake event for the STA after frame 602. Hence, both the RSSDT and the RTTDT sensors reported the anomaly and the Quite DoS attack was identified correctly and accurately.

In *Scenario One*, *Scenario Two*, *Scenario Three* and *Scenario Four*, as expected, the *RSSdiff* and *RTTdiff* values increased as the attacker was placed further away from the STA. In *Scenario Five* and *Scenario Six*, the AP, the IDS sensor and the attacker were located in close proximity of each other and as expected, the *RSSdiff* and *RTTdiff* values increased as the STA moved away from them and decreased as the STA moved closer. In *Scenario Seven* and *Scenario Eight*, the attacker was located further away from the IDS sensor and the AP. The observed *RTTdiff* and *RSSdiff* values increased as the STA moved away from the attacker, and decreased as it moved closer to the attacker (see Tables 1, 2, and 3).

An interesting observation was made that in *Scenarios Two*, *Scenario Three* and *Scenario Four*, where all parties were stationary; all the false positives were detected in frames generated after the attack had commenced (Tables 1 - 6). This means that all the false positives were caused by abnormal fluctuations in observed RSS and RTT values for the attacker. This observation was most likely the result of increasing distance between the attacker and the passive IDS monitor from *Scenario Two* to *Scenario Four*. Lack of line of sight connectivity and presence of various obstacles (walls, doors etc.) most likely acted as contributing factors to random fluctuations in observed RSS and RTT values for the attacker. Being positioned in close proximity of the sensor in all these scenarios, the STA did not suffer such random fluctuations and hence did not generate any false positives before the attack was launched.

However, in *Scenario five* to *Scenario Eight*, just the opposite was observed. The false positives were detected in frames generated before the attack had commenced, which meant that the source of these abnormalities was the STA and not the attacker. In these scenarios, the attacker was always stationary and the STA was in motion. These false positives can be attributed to the fluctuations in observed RSS and RTT values for the STA as a result of it being in motion.

The correlation technique successfully managed to keep the number of these false positives fairly low. The RSSDT and the RTTDT both successfully detected the performed attacks.

TABLE 1. True Positives for TKIP DoS Attack experiments

Scenario	**RSS diff**	**Frame number**	**RTT diff**	**Frame number**
One	NA	NA	NA	NA
Two	19	499	29.651	510
Three	34	550	39.204	585
Four	47	606	51.014	657
Five	32	554	43.751	590
Six	23	622	31.098	634
Seven	27	530	38.773	549
Eight	30	583	40.007	603

TABLE 2. True Positives for Channel Switch DoS Attack experiments

Scenario	**RSS diff**	**Frame number**	**RTT diff**	**Frame number**
One	NA	NA	NA	NA
Two	16	520	25.556	543
Three	37	603	40.002	630
Four	42	583	49.099	599
Five	27	532	40.071	545
Six	26	601	37.655	628
Seven	27	530	38.773	549
Eight	30	583	40.007	603

TABLE 3. True Positives for Quite DoS Attack experiments

Scenario	**RSS diff**	**Frame number**	**RTT diff**	**Frame number**
One	NA	NA	NA	NA
Two	16	602	26.447	620
Three	34	603	39.077	626
Four	38	593	47.014	639
Five	30	495	38.977	512
Six	28	598	36.007	620
Seven	22	617	31.055	633
Eight	26	589	32.411	603

Fig.s 3, 4, and 5 show the distribution of true positives and false positives registered by the RSSDT and the RTTDT when running the TKIP DoS Attack, the Channel Switch Attack, and the Quite Attack experiments respectively. It is clear from these Fig.s that the used techniques are effective in detecting the applied attacks because of the absence of false negatives.

Single Anomalies

A single anomaly would occur if a RSS alert was registered by the RSSDT, while the RTTDT did not register an anomaly in the next RTS-CTS event for that MAC address. Another example would be if a RTT alert was raised by the RTTDT but the next RSS reading for that MAC address was below the threshold. The RSSDT and the RTTDT only raise an alert if the difference between the last observed and current characteristic is above a threshold. Single anomalies were ignored by the correlation engine and an alarm was only raised if both the detection techniques register an alert.

Fig. 3, 4, and 5 show the distribution of single anomalies registered by the RSSDT and the RTTDT when running the TKIP DoS Attack, the Channel Switch Attack, and the Quite Attack experiments respectively, where the correlation engine did not raise any alarm.

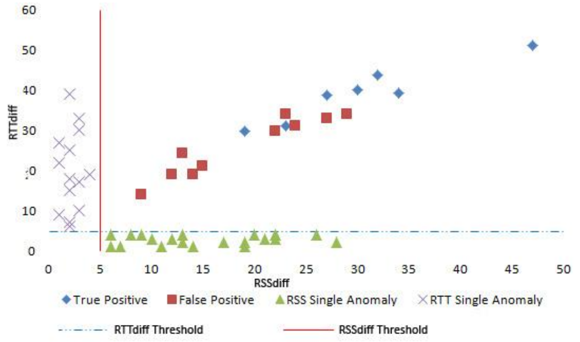

Figure 3. Alarms and Single Anomalies for TKIP DoS Attack Experiment

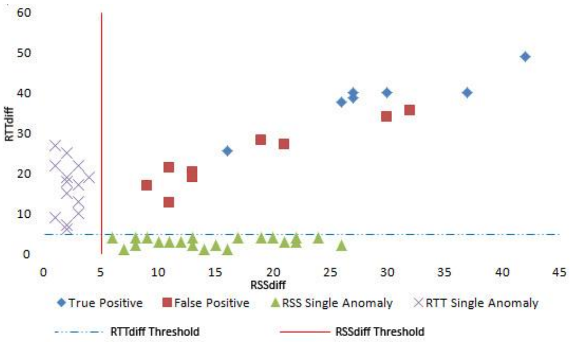

Figure 4. Alarms and Single Anomalies for Channel Switch DoS Attack Experiment

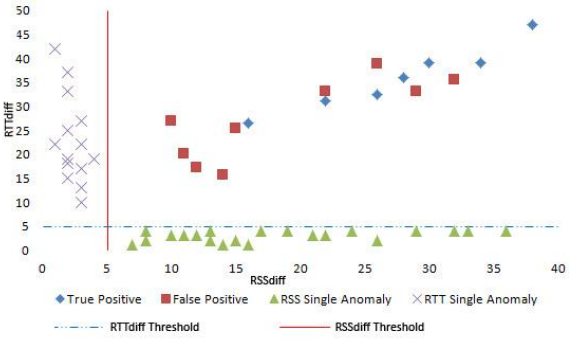

Figure 5. Alarms and Single Anomalies for Quite DoS Attack Experiment

TABLE 4. False Positives for TKIP DoS Attack experiments

Scenario	**RSS diff**	**Frame number**	**RTT diff**	**Frame number**
One	NA	NA	NA	NA
Two	14	730	19.011	744
Three	24 27	698 811	31.112 33.009	718 840
Four	23	800	34.093	832
Five	13	389	24.225	411
Six	12	370	18.999	393
Seven	15	278	21.112	307
	9	396	14.001	420
Eight	29 22	290 340	34.112 29.901	312 366

TABLE 5. False Positives for Channel Switch DoS Attack experiments

Scenario	**RSS diff**	**Frame number**	**RTT diff**	**Frame number**
One	NA	NA	NA	NA
Two	13	698	20.332	709
Three	30 32	721 826	33.878 35.552	754 865
Four	19	748	28.221	773
Five	11	335	21.222	378
Six	9	293	16.988	316
Seven	13 11	271 377	18.909 12.727	300 399
Eight	21	389	27.101	409

TABLE 6. False Positives for Quite DoS Attack experiments

Scenario	**RSS diff**	**Frame number**	**RTT diff**	**Frame number**
One	NA	NA	NA	NA
Two	11	765	19.991	789
Three	29 32	777 870	33.117 35.550	808 897
Four	26	781	38.887	800
Five	10	278	27.002	301

Six	15	390	25.333	409
Seven	22	410	33.011	437
Eight	12 14	300 378	17.112 15.553	329 404

VI. Threshold Optimization

In experiments discussed above the RSSdiff threshold and the RTTdiff threshold were set to an initial constant value of 5. This means a RSS anomaly was only registered if the RSSdiff was greater than 5 and a RTT anomaly was only acknowledged if the RTTdiff value was greater than 5. An alarm was raised by the correlation engine only when both the RSSdiff threshold and the RTTdiff threshold were exceeded. The threshold value 5 was thought to be just low enough to avoid a high number of false negatives and just high enough to avoid a large volume of false positives. Since ideally both the techniques should exhibit the same level of accuracy, the same threshold value was used for both. In these experiments (in sections 4.3 and 4.4), both the thresholds were set to the same value, in fact there is no need for the RTTdiff threshold and the RSSdiff threshold to be with the same value.

In reality, we need to optimize these threshold values to ensure the lowest possible number of false positives and false negatives. Choosing the best threshold value for each detection technique can be performed using the algorithm presented in Fig. 6.

Figure 6. RSSdiff threshold and RTTdiff threshold optimization algorithm

```
1. Read the captured packets from the dump file
2. Filter the points according to their class (true positive, false positive…)
3. Let $F_{min}$=($RSSdiff_{min}$, $RTTdiff_{min}$) be the frame of class “true positive” such that it has the least RSSdiff and RTTdiff values
4. If $RSSdiff_{min} \epsilon \mathbb{N}$
        Set the optimized RSSdiff threshold = $RSSdiff_{min} – 1$
   Else If $RSSdiff_{min} \epsilon \mathbb{R}$
        Set the optimized RSSdiff threshold = $\lfloor RSSdiff_{min} \rfloor$
5. If $RTTdiff_{min} \epsilon \mathbb{N}$
        Set the optimized RTTdiff threshold = $RTTdiff_{min} – 1$
   Else If $RSSdiff_{min} \epsilon \mathbb{R}$
        Set the optimized RTTdiff threshold = $\lfloor RTTdiff_{min} \rfloor$
   Where $\mathbb{N}$ is the set of all natural numbers, and $\mathbb{R}$ is the set
   of all real numbers
```

Table 7 represents the optimized thresholds after applying the algorithm in Fig. 6 on the results the experiments section. Referring to Table 7 we found that the optimized thresholds are very close for the three attacks experiments. In our opinion, from a general intrusion detection perspective, it is far more critical for an IDS to minimize the false negative rate than to maintain a low false positive rate. The cost of missing an attack is much higher than the cost of raising a false alarm. Therefore, we choose the minimum threshold to avoid the false negatives i.e. 15 for RSSdiff threshold and 25 for RTTdiff threshold. Fig.s 3, 4, and 5 represent the true positives, false positives and single anomalies registered by the IDS when using the initial threshold settings. In Fig.s 3, 4, and 5; RSSdiff Threshold and RTTdiff Threshold refer to initial values used for thresholds (i.e. 5 for all scenarios).

Fig. 7, 8, and 9 represent the true positives, false positives and single anomalies registered by the IDS when using the optimum threshold settings. In Fig. 7, 8, and 9, RSSdiff Threshold and RTTdiff Threshold refer to the optimized values of RSSdiff and RTTdiff thresholds respectively. These values were generated by applying the Algorithm in Fig. 6 to minimize the number of false positives and false negatives.

Table 7 demonstrates that RSSdiff threshold of 15 and RTTdiff threshold of 25 which are the optimum choice for the thresholds of the detection techniques. Fig.s 7, 8, and 9 show how the single anomalies, false positives and true positives are affected by the new optimum threshold values. As a result of the new thresholds, some false positives became RSS single anomalies or RTT single anomalies. The new thresholds did not introduce any false negatives since there are no true positives became false negatives. Moreover, no single anomaly was converted into a false positive as a result of the new threshold. In fact, some of the single anomalies (both RSS and RTT single anomalies) became normal events. Hence with 100% true positive detection, RSSdiff Optimized Threshold of 5 and RTTdiff Optimized Threshold of 25 is proved to be the optimum threshold values for the resented test scenarios.

TABLE 7. Optimized RSSdiff and RTTdiff thresholds

Experiment	**New *RSSdiff* threshold**	**New *RTTdiff* threshold**
TKIP DoS Attack	18	29
Channel Switch DoS Attack	15	**25**

Quite DoS Attack	<u>**15**</u>	26

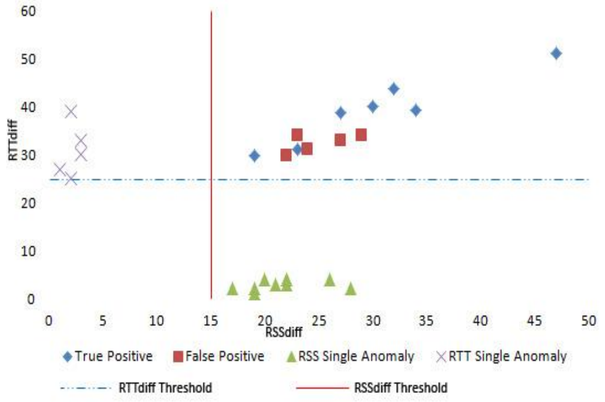

Figure 2. Alarms and Single Anomalies for TKIP DoS Experiment when applying the optimized threshold

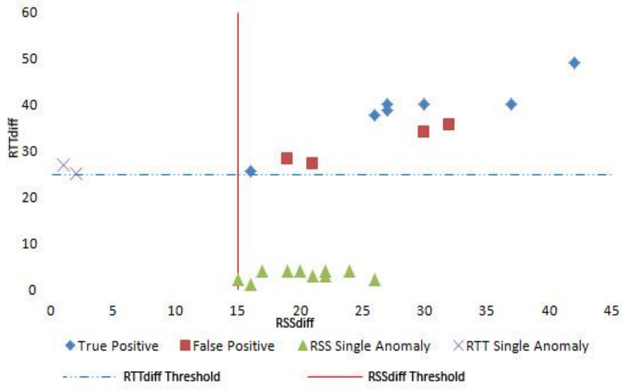

Figure 3. Alarms and Single Anomalies for Channel Switch DoS Experiment when applying the optimized threshold

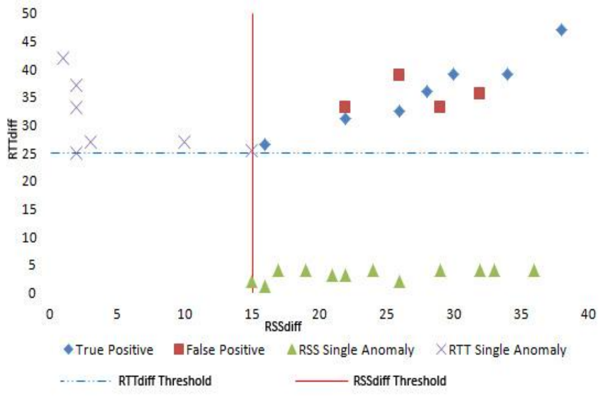

Figure 4. Alarms and Single Anomalies for Quite DoS Experiment when applying the optimized threshold

Accuracy and efficiency of the RSSDT and the RTTDT depends on the choice of suitable threshold values and hence places a large expectation on these threshold values to be optimally calculated. This increases the importance of our developed algorithm (Fig. 6).

Thresholds are unique to each WLAN environment and can also change frequently. Hence, the thresholds should be regularly calculated to optimum values. Using a distributed approach and deploying multiple distributed co-operating IDS sensors can decrease this expectation on the accuracy of the threshold values. Rather than relying on the alarms generated by a single IDS sensor, the intrusion detection process can be enhanced by correlating detection results across multiple sensors.

This also makes it a much harder job for the attacker to launch a successful spoofing attack as they will have to guess and spoof the RSS and the RTT values for the legitimate nodes, as observed by each IDS sensor. This will require the attacker to be at multiple locations at the same time, hence making it very hard for the attacker to launch an undetected attack.

VII. CONCLUSION

Despite using a number of preventative security measures, IEEE 802.11i RSNs still suffer from multiple vulnerabilities that can be exploited by an adversary to launch attacks. This underlines the need for using a monitoring framework as a second layer of defense for WLANs. Such monitoring capability can be implemented using a wireless intrusion detection system.

This paper verifies the effectiveness of RSSDT and RTTDT wireless intrusion detection techniques that address majority of RSN attacks. The paper proposed an algorithm to enhance the performance of the correlation of the Received Signal Strength Detection Technique (RSSDT) and Round Trip Time Detection Technique (RTTDT) in detecting MAC spoofing Denial of Service (DoS) attacks. The proposed algorithm is enhanced the performance by optimizing the value of the detection threshold.This paper also demonstrates that the detection results can be correlated across the WIDS sensors and also the detection techniques themselves to provide greater assurance in the reliability of the alarms and enable automatic attack scenario recognition.

The experiments presented in Section 2 demonstrate the feasibility of using RSS and RTT monitoring as wireless intrusion detection techniques since they did not produce any false negatives, while the correlation between the RSSDT and RTTDT and the self-adaptation for both RSSDT and RTTDT thresholds results was feasible in lowering the number of false positives.

REFERENCES

[1] B. Aboba, L. Blunk, J. Vollbrecht, J. Carlson, and H. Levkowetz. Extensible Authentication Protocol (EAP). The Internet Engineering Task Force-Request for Comments, RFC 3748, 2004.

[2] R. Ahlawat and K. Dulaney. Magic Quadrant for Wireless LAN Infrastructure, 2006. Gartner Research, 2006.

[3] J. Anderson. Computer Security Threat Monitoring and Surveillance. Report 79F296400, James P.Anderson Co., 1980.

[4] P. Bahl and V. Padmanabhan. RADAR: an in-building RF-based user location and tracking system. INFOCOM 2000. Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE, 2, 2000.

[5] J. Ellch. Fingerprinting 802.11 Devices. PhD thesis, Naval Postgraduate School; Available from National Technical Information Service, 2006.

[6] J. Franklin, D. McCoy, P. Tabriz, V. Neagoe, J. Van Randwyk, and D. Sicker. Passive Data Link Layer 802.11 Wireless Device Driver Fingerprinting. Proceedings of the 15th Usenix Security Symposium, 2006.

[7] R. Gill, J. Smith, and A. Clark. Experiences in Passively Detecting Session Hijacking Attacks in IEEE 802.11 Networks. In R. Safavi-Naini, C. Steketee, and W. Susilo, editors, Fourth Australasian Information Security Workshop (Network Security) (AISW 2006), volume 54 of CRPIT, pages 221–230, Hobart, Australia, 2006. ACS.

[8] R. Gill, J. Smith, M. Looi, 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 (AusCert 2005): Refereed R&D Stream, pages 26 – 38, 2005.

[9] S. Glass and V. Muthukkumarasamy. A Study of the TKIP Cryptographic DoS Attack. 15th IEEE International Conference on Networks, 2007.

[10] A. Godber and P. Dasgupta. Countering rogues in wireless networks. International Conference on Parallel Processing Workshops, pages 425–431, 2003.

[11] F. Guo and T. Chiueh. Sequence Number-Based MAC Address Spoof Detection. In 8th

[12] IEEE. 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 Amendment 6: Medium Access Control (MAC) Security Enhancements, 2004. Institute of Electrical and Electronics Engineers.

[13] V. Kasarekar and B. Ramamurthy. Distributed hybrid agent based intrusion detection and real time response system. First International Conference on Broadband Networks (BroadNets 2004), pages 739–741, 2004.

[14] B. Konings, F. Schaub, F. Kargl, and S. Dietzel. Channel switch and quiet attack: New DoS attacks exploiting the 802.11 standard. IEEE 4th Conference on Local Computer Networks, 2009.

[15] D. Madory. New Methods of Spoof Detection in 802.11 b Wireless Networking. PhD thesis, Dartmouth College, 2006.

[16] NIST. Establishing Wireless Robust Security Networks: A Guide to IEEE 802.11i, Feb 2007. National Institute of Standards and Technology. Special Publication 800-97. [Online] Available: http://csrc.nist.gov/publications/ nistpubs/800-97/SP800-97.pdf

[17] A. Smailagic and D. Kogan. Location sensing and privacy in a context-aware computing environment. Wireless Communications, IEEE Personal Communications, 9(5):10–17, 2002.

[18] Tkiptun-ng, 2010, [Online] Available: http://www.aircrack-ng.org

[19] NSA. Guidelines for the Development and Evaluation of IEEE 802.11 Intrusion Detection Systems (IDS), Nov 2005. National Security Agency. Network Hardware Analysis and Evaluation Division.[Online] Available:http://www.nsa.gov/notices/notic00004.cfm?Address=/snac/wireless/I332-005R-2005.pdf

[20] R. Bace and P. Mell. NIST Special Publication on Intrusion Detection Systems. National Institute of Standards and Technology, draft document, February, 2001.

[21] H. Debar and J. Viinikka. Intrusion detection: Introduction to intrusion detection and security information management. FOSAD, 2005:2005, 2004.