Detecting Identity Spoofs in IEEE 802.11e Wireless Networks

Gayathri Chandrasekaran*, John-Austen Francisco*, Vinod Ganapathy*, Marco Gruteser†, Wade Trappe†,

*Dept. of CS, Rutgers University
North Brunswick, NJ 08902
{*chandrga,deymious,vinodg*}@*cs.rutgers.edu*

†WINLAB, Rutgers University
Piscataway, NJ 08854
{*gruteser,trappe*}@*winlab.rutgers.edu*

***Abstract*—Wireless networks are vulnerable to identity spoofing attacks, where an attacker can forge the MAC address of his wireless device to assume the identity of another victim device on the network. Identity spoofing allows an attacker to avail network services that are normally restricted to legitimate users. Prior techniques to detect such attacks rely on characteristics such as progressions of MAC sequence numbers. However, these techniques can wrongly classify benign flows as malicious with newer 802.11e wireless devices that allow multiple progressions of MAC sequence numbers from the same device. Several other techniques that rely on physical properties of transmitting devices are ineffective when the attacker and the victim are mobile. In this paper, we propose an architecture to robustly detect identity spoofing attacks under varying operating conditions. Our architecture employs a series of increasingly powerful detectors to identify or eliminate the possibility of an attack, culminating in a powerful, RSSI-based per-packet localizer that reliably detects identity spoofing attacks. We implemented this architecture and used it to detect a variety of identity spoofing attacks. Our experiments show that it can effectively detect identity spoofs with a low false positive rate of 0.5%.**

I. Introduction

As 802.11 networks increase in popularity for connecting to the Internet, so also have attacks against such networks. One such class of attacks are *identity spoofing* attacks, where a malicious user attempts to acquire the identity of a legitimate user of an 802.11 wireless network. Masquerading as a legitimate user allows the malicious user to avail of services that are normally restricted to legitimate users of the wireless network. Because MAC addresses associated with wireless cards are often used to identify individuals in 802.11 networks, masquerading attacks typically work by spoofing MAC addresses.

Identity spoofing attacks can potentially be overcome by using cryptographic techniques instead of MAC addresses to manage identity. However, cryptographic authentication of devices introduces key management overheads that may not be practical for several commodity wireless networks. Consequently, the research community has recently sought several non-cryptographic solutions to detect identity spoofing attacks using communication properties unique to wireless protocols being employed. For example, MAC and physical layer parameters, such as the frame sequence number, received signal strength (RSS) and probe frequency have recently been examined as a possible means to identify wireless devices. However, as discussed below, each of these schemes has drawbacks that either lead to a high false positive rate or fail to detect attacks.

Techniques that use MAC sequence numbers for detection find spoofed identities when multiple MAC sequence progressions are associated with the same MAC number [10], [11], [8]. MAC sequence numbers increase linearly (typically in increments of one); these techniques report an attack when a MAC sequence numbers observed fail to exhibit monotonic behavior. Unfortunately, these techniques report high false positive rates with 802.11e-enabled wireless devices, which support multiple transmission queues, thus allowing multiple legal MAC sequences to be associated with the same wireless device. To address these shortcomings, recent work has explored the use of parameters from the physical layer for spoof detection [6], [4], [15]. For example, these techniques observe RSS values associated with packets measured at one or more receiver antennas. The RSS values are correlated with transmission power, the separation between the transmitter and the receiver, and the complexity of the radio environment in which communication takes place. These techniques detect identity-based attacks by determining that a MAC address is associated with multiple locations. However, these techniques are typically effective only in static settings, as it is well-known that RSS values can oscillate even in non-adversarial settings with legitimate users who are mobile. In such scenarios, RSS and other physical layer parameter-based solutions thus result in a large number of false positives.

In this paper, we propose a robust architecture for identity spoof detection that overcomes the limitations of prior work. We first study the characteristics of a wide range of 802.11 wireless devices and show that identity spoof detection techniques that employ a single metric for detection and those that do not account for operating conditions, such as device types and the dynamics of associated wireless devices, will likely fail. We then present a layered architecture that employs a series of detectors (increasing in complexity) to robustly detect identity spoofing in both mobile and static scenarios and in networks with 802.11e-enabled devices.

Intuitively, this architecture employs a process of elimination: a cheap detector observes wireless traffic to eliminate the possibility of an attack or to confirm an attack; if it cannot conclusively deduce the possibility or absence of an attack, it invokes a costlier detector. In particular, we first classify incoming packets using their MAC sequence number and packet type, and determine whether sequence numbers associated with each packet type increase linearly. Detecting identity spoofs using this technique is cheap, but can result in false positives upon the use of QoS streams to transmit packets (as allowed by the 802.11e standard). Consequently, our detection architecture further analyzes anomalous traffic to localize each network packet and determine the Euclidean distance between them. If the distance is above a certain threshold that we empirically evaluate, it reports an identity spoof attack.

To summarize, the main contributions of this paper are:

- **A study of characteristics of 802.11 wireless devices.** We analyze a wide range of wireless devices and show that techniques that use a single metric to detect identity spoofs will likely fail.
- **A robust detection architecture.** We propose an architecture to detect identity spoofing. Our architecture robustly and precisely detects identity spoofs even with 802.11e-enabled devices.
- **Experimental evaluation.** We evaluate our architecture using the ORBIT [14] platform and show that it can robustly detect attacks with low false positive rates.

II. Related Work

There is much prior work on detecting identity spoofing attacks; Bellardo and Savage [2] provide a detailed summary of such attacks. We restrict our discussion of related work to masquerading attacks, where an attacker assumes the identity of the victim. We discuss three classes of defenses against masquerading attacks.

Analysis of MAC sequence numbers. The MAC header of every data and management frame has a twelve bit sequence number field assigned by the MAC layer; this number is incremented for each subsequent frame. When an attacker takes the identity of a victim, frames captured from the attacker would have a different sequence number progression as compared to that of the victim. Prior work [10], [11], [8] has used this idea to use "oscillations" in sequence numbers as an indicator of an attack. However, this technique will report false positives with newer, 802.11e-enabled wireless devices because the 802.11e standard allows nine legal sequence number progressions from a single device. The architecture proposed in this paper robustly handles 802.11e devices without reporting false positives.

Analysis of RSS. The received signal strength (RSS) is a measure of the power present in a radio signal. RSS is related to transmission power, the separation between the transmitter and the receiver, and the radio environment in which communication happens. Detection techniques that use RSS are based upon the intuition that an attacker who is not geographically close to a victim is likely in a different radio environment. This would lead to a difference in the RSS values observed for packets originating from the attacker and the victim. Faria and Cheriton [6] detect spoofing using the RSS fingerprint oscillations observed at multiple receivers. Chen et al. [4] observed RSS values for every MAC address and perform k-means clustering on these values. A cluster centroid separation of 6db or higher was an indication of an attack. However, neither approaches can robustly handle mobile victims; in this case, RSS values oscillate even if there is no attack. Our architecture handles masquerading attacks in mobile environments with very low false positives.

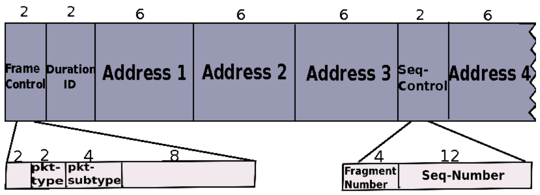


Fig. 1. Components of a MAC frame.

Device Profiling. Sheng et al. [15] build a Gaussian Mixture Model (GMM) for each transmitter at every receiver using observed RSS values. Under attack, the observed distribution fails to fit the profiled GMM. However, when the victim is mobile, the RSS profiling scheme is ineffective. Franklin et al. [7] profile the frequency of probe request packets from all clients using a sudden increase in the probe request frequency the card to detect the presence of multiple devices under the same identity. Although this scheme is robust to victim mobility, commodity cards with Madwifi drivers [12] can possibly disable probing, thereby making an attack undetectable using this scheme. Other techniques for fingerprinting devices include studying minute imperfections induced at the manufacturer, which may manifest as clock skews [9] or as artifacts of emitted signals [3] These techniques are complementary to the work proposed in this paper, which does not use profiling.

Table I summarizes related work and shortcomings of existing defenses.

III. Detecting Identity Spoofs in 802.11e Wireless Networks

This section presents a study of 802.11 device characteristics and the design of our detection architecture. We begin by discussing the threat model.

We assume that the attacker and the victim use devices with standard off-the-shelf wireless transmitters (with or without QoS provisioning); we place no restrictions on the antennas of these transmitters (they could either be omni-directional or directional). The attacker is free to manipulate the MAC address and the tranmission power. However, we assume that the (benign) victim does not modify MAC header parameters (which is typically the case). Our detection infrastructure allows both the attacker and victim to be mobile within the area covered by the landmarks of our localization infrastructure. Both the attacker and the victim must be present in the network and must be active simultaneously for detection to succeed.

A. Wireless device characteristics

The structure of a MAC frame of a packet transmitted from an 802.11 wireless device is shown in Figure 1. As this Figure shows, the MAC frame contains a two byte *frame control field*, a two byte *duration ID field*, a two byte *sequence control field* and four *address* fields, each six bytes long.

Defense	Technique	Shortcoming
RSS values [6], [4]	Use RSS variations to detect attack	Victim mobility induces false positives
MAC sequence number [8], [10], [11]	Check for linear progressions of MAC sequence numbers	Not Compatible with 802.11e enabled cards
Device profiling [15], [7]	Profile RSS distributions and probe frequencies of victim; use anomaly detection	Victim mobility induces false positives; attacker can disable probing

TABLE I
SUMMARY OF DEFENSES AGAINST MASQUERADING ATTACKS

These four address fields are used to indicate the basic service set identification (BSSID), source address (SA), destination address (DA), transmitted STA address (TA) and receiving STA address (RA). The locations of these address fields inside the MAC frame depend on the type of the frame.

The sequence control field in a 802.11 MAC header is a two byte field and itself consists of a four bit *fragment number* and a twelve bit *sequence number*. Each higher level frame is assigned a sequence number field as it is passed to the MAC for transmission. The sequence number subfield operates as a counter (mod 4096) and is incremented by one for subsequent frames. However, the sequence number does not change for subsequent fragments of a fragmented packet; instead the fragment number is incremented. The frame control field of the MAC header is another two byte field, which includes a two bit type field and a four bit packet subtype field. The packet can either be a management, control or data packet, as indicated by the type field; data frames are further classified into regular data frames or QoS data frames. Together, the type and subtype fields of the MAC header define the packet type. Both the packet type and its sequence number can be extracted from the frame control field and the sequence control field, respectively, at the MAC layer.

Newer 802.11e-enabled commodity wireless cards have QoS extensions, where assumptions about monotonicity of sequence numbers in packets originating from a device do not typically hold. As indicated in the 802.11e standard [1], every QoS-enabled station associated with a QoS-enabled AP maintains one mod 4096 counter per QoS priority class for each receiver. There are eight QoS priority classes (numbered from zero to seven), and more than one QoS stream could be active from the QoS-STA at any point. All non-QoS data frames and the management frames sent by a QoS-enabled station are assigned a sequence number using an additional module 4096 counter. Thus, there can be upto *nine* legal simultaneous sequence number progressions from any single source. It is therefore necessary for an identity spoof detection technique to classify packets based on sequence number and packet type per QoS priority class; failing this, the detection technique would report false positives, i.e., it would classify benign packets as possibly identity-spoofed packets.

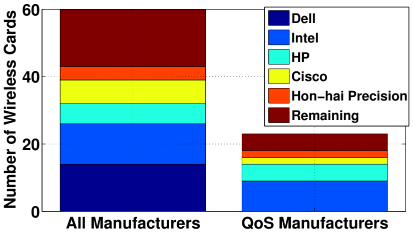


Fig. 2. 802.11 wireless devices in our testbed.

Type	# of Devices	Percentage
non-QoS devices	37	61.66%
QoS devices with 1 active queue	20	33.33%
QoS devives with $>$ 1 active queue	3	5%

TABLE II
DEVICES WITH QOS PROVISIONING ENABLED.

To better understand wireless transmission characteristics of 802.11 devices and the impact of newer 802.11e standards on detection accuracy, we profiled a typical office environment on a busy afternoon to study network statistics. Our test environment had 60 wireless cards, and as shown in Figure 2, were manufactured by a variety of vendors. Of these 60 cards, 23 (nearly 40%) had QoS provisioning enabled. Of these 23 cards, three actively communicated with packets belonging to different QoS priority classes. Table II summarizes our findings. This study shows that identity spoof detection techniques that use MAC sequence numbers alone would report false positives on the cards that have QoS provisioning enabled, which account for nearly 40% of the cards in our test environment.

B. Detection metrics

Motivated by the above discussion, we chose the *MAC sequence number, classified according to packet type and QoS priority class* as one of the metrics in our identity spoof detection algorithm. For backwards compatibility, we also monitor monotonic sequence number progressions from non-QoS-enabled stations without classifying them based on packet type, priority or receiver. While this metric alone serves to correctly classify a large fraction of network traffic as anomalous or benign, this metric fails to determine whether two packets that belong to the same source MAC address and different QoS priority classes came from the same device. This is because an attacker could choose to transmit QoS packets with a priority class that is unused by the victim's card, thereby allowing the identity spoof attack to proceed undetected.

To verify that frames that use differentiated services with the same source MAC address originate from the same device, we additionally monitor the locations from which packets belonging to these different priority classes originate using the *Euclidean distance in location space between packets*.

Euclidean distance between packets. The *received signal strength indicator* (RSSI) is a measure of the power present in a received radio signal. When a wireless receiver is placed in monitor mode, it can determine the RSSI for *all* packets being transmitted within the range of the receiver. By aggregating per-packet RSSI captured at different receivers, it can generate a RSSI fingerprint, which is a vector of RSSI values—one from each receiver. Because receivers are not time-synchronized, per packet RSSI aggregation requires a unique packet identifier and the following tuple, which serves as a unique packet identifier.

<Sender MAC, MAC seq #, packet type, recv. timestamp>

RSSI values from different receivers that have the same sender MAC, MAC sequence number, packet type and "close-enough" timestamp are aggregated to produce a per-packet fingerprint. We assume the existence of a localization infrastructure, equipped with multiple landmarks to capture packets and their RSSI values. If a localization landmark does not have the trace of a packet that other landmarks have captured, the aggregation process uses a default value of -99. Localization infrastructures determine the location of a packet based upon the RSSI values observed for the packet.

Assume that localizing a packet P_i yields the coordinates (X_i, Y_i). The *differential Euclidean distance* between packet P_i and P_{i+1} is the Euclidean distance between (X_i, Y_i) and (X_{i+1}, Y_{i+1}). In a scenario without identity spoof attacks, the differential Euclidean distance between successive packets from a single source (whether static or mobile) should be within a small threshold; in particular, it should be bounded above based upon the speed of the device and the time interval between measurements. However, under an identity-spoof attack where the attacker and the victim are geographically separated, the differential Euclidean distance between the attacker's packet and the victim's packet will be much larger. This metric is highly effective at detecting identity spoofs except when the attacker and victim are extremely close geographically, and both use the same power levels and similar wireless cards. In this case, the attacker and victim might be localized to the same location, thereby bypassing detection.

C. Detection Algorithm

Our detection algorithm (Algorithm 1) works on data collected using commodity wireless cards in monitor mode; these wireless cards form part of our detection infrastructure. In this mode, the card cannot transmit but can receive all data sent on the channel.

The detection algorithm operates on a sequence of wireless packets captured by this infrastructure. It first obtains the MAC sequence numbers of packets associated with a source MAC address and checks whether the sequence numbers increase in linear progression. If this is the case, it concludes there is no attack. However, sequence numbers may not be in linear progression because of retransmissions and interference. The detection algorithm accounts for these distortions; as long as distortions are within acceptable thresholds, the algorithm will not declare that an attack is in progress.

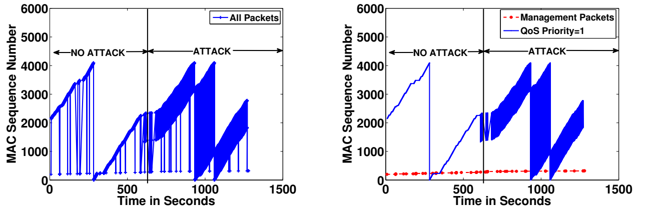
```
Algorithm: Find-Identity-Spoofs
Input     : S: Sequence of wireless packets
Output    : Attack/No_Attack
MACs = list of MAC sequence numbers in S;
if MACs in linear progression then
    return No_Attack;
else if MAC variation in valid range then
    return No_Attack;
/* MAC sequences not in linear
   progression; check frame types      */
FTypes = frame types extracted from S;
if FTypes $\in$ {Management, Regular Data} then
    return Attack;
/* Frame type must be QoS-Data; examine
   priorities                            */
QoS-Priorities = QoS priorities extracted from S;
if QoS-Priorities are all the same then
    return Attack;
/* QoS priorities are either mixed, or
   mixed QoS-data and regular data     */
Perform differential localization for packets in S;
if Euclidean distance between successive packets exceeds
threshold then
    return Attack;
return No_Attack;
```

Algorithm 1: Algorithm to detect identity spoofs in 802.11e-enabled wireless networks.

If MAC sequence numbers are not in linear progression, the detection algorithm checks packet types to determine whether the frames are data frames. Because management and regular data frames are associated with a single MAC sequence counter, they must be in linear progression; therefore the algorithm terminates (with "Attack in progress") if the frames are management or data frames. QoS-Data frames must be analyzed further to determine whether an attack is in progress. The algorithm thus reparameterizes the QoS-data frames based upon their QoS priorities and each series of frames within a given priority are examined. This step is necessary because each QoS priority level can use a different sequence counter per receiver. If all QoS priorities in the sequence of packets are the same, then this is indicative of an attack, because MAC sequence numbers within the same QoS priority level must normally be in linear progression.

If QoS priority levels differ, the algorithm cannot conclusively determine whether an attack is in progress or not *even if MAC sequence numbers within each QoS priority level are in linear progression*. This is because an attacker masquerading as a victim could transmit packets on a QoS priority level that is unused by the victim. To determine the possibility of an attack, we employ differential localization of *each* packet in the sequence. Using supervised Bayesian learning techniques [5] we determine the location each packet was emitted from and compute the differential Euclidean distance between successive packets. If this distance exceeds an acceptable threshold, the algorithm declares a possible attack.

Security analysis. An attacker masquerading as a victim will transmit network packets with the MAC address of the victim. Because the MAC sequence numbers of the victim's



(a) sequence number classifier

(b) sequence number, packet type and QoS priority class Classifier

Fig. 3. Packet from Victim Classified according to (a) MAC Sequence Number (b) MAC Sequence Number, Packet Type and QoS Priority

packets follow a linear progression, the attacker's packets will result in multiple progressions of MAC sequence numbers. Note that multiple progressions alone will trigger false positives in previously proposed schemes [10], [11], [8]. To avoid such false positives in the presence of multiple linear progressions of MAC sequence numbers, Algorithm 1 further filters packets by their frame types. Only QoS-data frames in 802.11e-enabled devices are associated with multiple counters and can thus have multiple linear progressions of MAC sequence numbers even when an attack is not in progress. Algorithm 1 therefore determines whether the frames are QoS-data frames; if not, it can declare an attack. For QoS-data frames, Algorithm 1 further determines whether QoS priorities are the same. Because a masquerading attacker can possibly transmit data packets using unused QoS priority levels, Algorithm 1 finally uses localization to identify whether an attack is in progress.

IV. Evaluation

We conducted experiments to evaluate the effectiveness of our detection algorithm in a typical office environment with partitioned cubicle offices. We used the open access ORBIT nodes [14] [1] to setup IEEE 802.11g wireless receivers at four different locations inside an office space. The wireless receivers were configured to monitor channel 11 in promiscuous mode.

We used a pair of laptops, one each as the attacker and the victim, each equipped with QoS enabled 802.11g card as the transmitters. The victim's wireless card was Intel-3945ABG and the attacker's was Netgear-WG511T. The wireless cards generated ICMP ping packets on channel 11 at the rate of 10 packets per second. We ran the Tshark packet sniffer utility at each of the receiver to capture the packets from the two transmitters. For each packet, we logged the transmitter's MAC address, the receiver's MAC address, MAC sequence number, RSSI and the time when the packet was captured. We then post-processed the packet traces to generate a per-packet fingerprint.

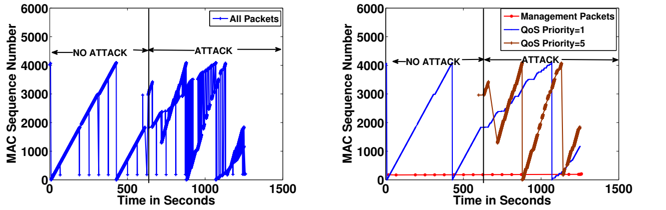
Two of the authors carried one laptop each and conducted two experiments - Same-priority masquerading attack and Different-priority masquerading attack. Each experiment lasted for twenty minutes, where there was no attack in the first ten minutes and there was a spoofing attack in the next ten minutes. In the first five minutes the victim was stationary whereas in the second five minutes, the victim was mobile. In the third five minutes, the victim was stationary and the attacker sat close (about 2 feet apart) to the victim and spoofed his MAC addresses. In the last five minutes, the attacker and victim were separated by large distances and the attacker spoofed the victim's identity. Throughout, both the attacker and the victim transmitted ping packets at the rate of 10 packets per second.

A. *Same-priority masquerading attacks*

In the same priority masquerading attack, the attacker changes his MAC address to that of the victim's MAC and transmits packets using the same QoS priority class with which the victim is transmitting. Figure 3 represents this attack, both as perceived by a MAC sequence number-based detection algorithm and our detection algorithm. As shown in Figure 3(a), when the victim transmits QoS and management packets, multiple progressions of MAC sequence numbers may exist; a technique that relies solely on MAC sequence numbers to detect attacks would report false positives even in scenarios where there is no attack because the management packets have a different sequence progression compared to the QoS packets. In contrast, Figure 3(b) which classifies the packets based on sequence number per packet type and QoS priority class eliminates this false positive. During an identity spoof attack, the sequence number scheme identifies an oscillation of sequence numbers for the packets belonging to the same MAC, packet type and QoS Priority class. This oscillation is correctly identified as an attack.

B. *Different-priority masquerading attacks*

In different priority masquerading attacks, an attacker not only changes his MAC address to that of the victim, but also ensures that the packets that he transmits belong to a different QoS priority class. As shown in Figure 4(b), under the different priority masquerading attack, the sequence number with packet type and priority class classifier checks for in-sequence progression within each QoS type and classifies the resulting traffic as legal. However, the two QoS streams here were caused by two *different* wireless cards (that of the victim and the attacker) having the same MAC address and operating with different QoS priority classes. Because it is not possible to determine whether the two QoS streams originated from the same device or from two different devices, we localize



(a) sequence number classifier

(b) sequence number, packet type and QoS priority class classifier

Fig. 4. Packet from victim classified according to (a) MAC Sequence Number (b) MAC Sequence Number, Packet Type and QoS Priority

[1] We emphasize that our experiments were not on the controlled 400 node ORBIT grid

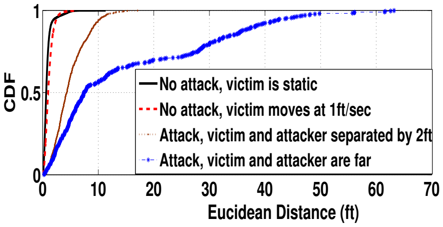


Fig. 5. Euclidean distance between successive packets.

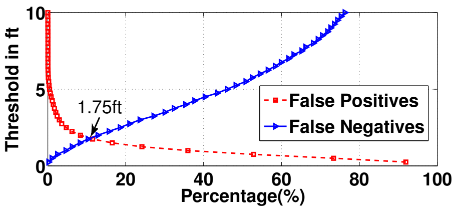


Fig. 6. Trade off between false positive and false negative rates for different Euclidean distance thresholds.

successive packets that belong to different priority classes using a Bayesian solver [5] and plot the Euclidean distance between those successive packets in Figure 5 (the last step of Algorithm 1).

Figure 5 shows that when there is no attack, i.e., when packets originate from the same device, the per packet Euclidean Distance falls within two feet 95% of time irrespective of whether the victim is stationary or mobile. While it is not surprising to find that the Euclidean distance between successive packets from a stationary victim is small, it is very encouraging to find that the Euclidean distance between successive packets from the mobile victim to be lesser than two feet (the second five minutes of our experiment). With the victim moving at a speed of one foot per sec, and transmitting at the rate of 10 packets per second, the distance that the victim could have moved between packet transmissions is approximately 0.1 feet. For this small displacement, the observed signal strengths at different receivers for a packet does not vary much as compared to that of the previous packet; consequently, the second packet is localized to a position close to that of the previous packet. It is important to note that the resulting euclidean distances is a function of the victim's speed and packet transmission rate.

Under attack, packets originate from two sources (attacker and the victim) with the victim's MAC address. We can see that when the attacker was sitting two feet apart from the victim, the resulting distance between successive packets vary from 0 to 15 feet with a median distance of five feet. This is because, at a two feet separation, the small scale fading experienced by the signals from the two transmitters that are separated by two feet could be completely different, resulting in signal strength changes that can be as large as 20db [13], which in turn affects the resulting location estimation. Similarly, when the attaker and victim are separated by larger distances, the resulting Euclidean distance between successive packets vary wildly between $0-65ft$ thereby showing that, it is possible to use the euclidean distance metric to determine whether different QoS streams originate from the same source or from multiple sources.

C. Determining Euclidean distance threshold

In this section, we empirically determine the Euclidean distance threshold that should be used for distinguishing attack traffic from genuine traffic. We do so by studying the tradeoff between false positives and negatives for various Euclidean distance thresholds, as shown in Figure 6. As we can see from the Figure, a threshold value of 1.75 feet yields equal false positive and negative rates of 10%. The high false positive rate is as a result of the attacker and victim being geographically close. From Table II, we recall that the classifier that uses sequence number with packet type and QoS priority reduces the false positive in the network from 40% to 5% and with the addition of the Euclidean distance threshold verification, the overall false positive further reduces to 0.5%

V. Summary

Prior techniques to detect masquerading attacks in wireless networks produce false positives with newer 802.11e wireless devices and in scenarios when attackers and victims are mobile. Motivated by the results of a study of several 802.11 wireless devices, we proposed an algorithm that robustly detects identity spoofs in 802.11e wireless networks with very low false positive rate.

References

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

[2] J. Bellardo and S. Savage. 802.11 denial-of-service attacks: Real vulnerabilities and practical solutions. In *USENIX Security Symposium*, August 2003.

[3] V. Brik, S. Banerjee, M. Gruteser, and S. Oh. Wireless device identification with radiometric signatures. In *Mobicom*, Sep 2008.

[4] Y. Chen, W. Trappe, and R. P. Martin. Detecting and localizing wireless spoofing attacks. In *IEEE SECON*, oct 2007.

[5] E. Elnahrawy, X. Li, and R. P. Martin. The limits of localization using signal strength: A comparative study. In *IEEE SECON*, pages 406–414, oct 2004.

[6] D. Faria and D. R. Cheriton. Detecting identity-based attacks in wireless networks using signalprints. In *ACM Workshop on Wireless Security (WiSe)*, September 2006.

[7] 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 *USENIX Security*, 2006.

[8] F. Guo and T. cker Chiueh. Sequence number-based mac address spoof detection. In *Proceedings of 8th International Symposium on Recent Advances in Intrusion Detection(RAID)*, September 2005.

[9] T. Kohno, A. Broido, and K. C. Claffy. Remote physical device fingerprinting. In *IEEE Symposium on Security and Privacy*, May 2005.

[10] Q. Li and W. Trappe. Relationship -based detection of spoofing -related anomalous traffic in ad hoc networks. In *IEEE SECON*, pages 50–59, September 2006.

[11] D. Madory. New methods of spoof detection in 802.11 wireless networking. Master's thesis, Dartmouth College, Hanover, New Hampshire, June 2006.

[12] MADWiFi. *Multiband Atheros Driver for WiFi*. http://madwifi.org, 2007.

[13] P. Nepa, G. Manara, S. Mugnaini, G. Tribellini, S. Cioci, G. Albasini, and E. Sacchi. Differential planar antennas for 2.4/5.2 ghz wlan applications. In *IEEE Int. Symposium of Antennas and Propagation Society*, pages 973–976, July 2006.

[14] D. Raychaudhuri and et.al. Overview of the orbit radio grid testbed for evaluation of next-generation wireless network protocols. In *Proc. IEEE WCNC*, volume 3, pages 1664–1669, March 2005.

[15] Y. Sheng, K. Tan, G. chen, D. Kotz, and A. Campbell. Detecting 802.11 mac layer spoofing using receiver signal strength. In *IEEE INFOCOM*, 2008.