This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE Secon 2009 proceedings.

Detecting Spoofing Attacks in Mobile Wireless Environments

Jie Yang*, Yingying Chen* and Wade Trappe†
*Dept. of ECE, Stevens Institute of Technology
Castle Point on Hudson, Hoboken, NJ 07030
{jyang, yingying.chen}@stevens.edu
† WINLAB, Rutgers University
North Brunswick, NJ 08854
trappe@winlab.rutgers.edu

***Abstract*—The flexibility and openness of wireless networks enables an adversary to masquerade as other devices easily. Identity-based spoofing attacks are serious network threats as they can facilitate a variety of advanced attacks to undermine the normal operation of networks. However, the existing mechanisms can only detect spoofing attacks when the victim node and the spoofing node are static. In this paper, we propose a method for detecting spoofing attacks in the mobile wireless environment, that is when wireless devices, such as the victim node and/or the spoofing node are moving. We develop the DEMOTE system, which exploits Received Signal Strength (RSS) traces collected over time and achieves an optimal threshold to partition the RSS traces into classes for attack detection. Further, our novel algorithm alignment prediction (ALP), when without the knowledge of spatial constraint of the wireless nodes, utilizes temporal constraints to predict the best RSS alignment of partitioned RSS classes for RSS trace reconstruction over time. Our approach does not require any changes or cooperation from wireless devices other than packet transmissions. Through experiments from an office building environment, we show that DEMOTE achieves accurate attack detection both in signal space as well as in physical space using localization and is generic across different technologies including IEEE 802.11 b/g and IEEE 802.15.4.**

I. Introduction

As computing and networking are shifting from the static model of the wired Internet toward the new and exciting "anytime-anywhere" service model of the mobile Internet, wireless systems will become increasingly programmable, interfacing with converged devices, and supporting new mobile applications. One serious class of threats that will affect the successful deployment of mobile wireless technologies are spoofing attacks. Spoofing attacks can be launched with little effort. The reason stems from the shared nature of the wireless medium, where adversaries can perform passive monitoring of useful identity information and then masquerade as another device using the collected identity.

Spoofing attacks can facilitate a variety of advanced attacks to significantly impact the normal operation of wireless networks [1]–[4]. Spoofing attacks on mobile wireless devices may further inflict security and privacy damages on the social life of the individual who carries wireless devices. There has been active work in detecting spoofing attacks [3], [5], [6]. [3] proposed the use of matching rules of Received Signal Strength (RSS) for spoofing detection, [5] used K-means cluster analysis of RSS, and [6] modeled RSS readings as a Gaussian mixture model to capture antenna diversity. However, these mechanisms only work in static wireless environments, i.e., the victim node has a fixed location. In this work, we focus on spoofing attack detection in mobile wireless environments, that is, the wireless devices including the victim node and/or the spoofing node are moving around. Thus, detecting spoofing attacks in mobile wireless networks is important as it allows the network to further exploit a wide range of defense strategies, and consequently helps to ensure secure and trustworthy communication in emerging mobile applications.

We propose a system called DEMOTE, detecting mobile spoofing attacks in wireless environments, which exploits the correlation within the RSS trace based on each node's identity to perform attack detection in either the signal space or the physical space. DEMOTE utilizes an unsupervised thresholding approach to find an optimal threshold to partition the RSS trace of a node identity into two classes. Given the RSS is distinctively correlated to a wireless node's physical location, the partitioned two classes will be highly correlated if there is no spoofing attacks, whereas less or not correlated when a spoofing attack is present.

The key challenge in DEMOTE is to reconstruct the RSS trace based on the partitioned classes that belong to different physical nodes accurately when a spoofing attack is occurring. Although we are lack of knowledge of knowing the spatial constraint of mobile wireless nodes, we found that there is a temporal constraint that is unique to the RSS trace from each physical node. We developed a simple algorithm, ALignment Prediction (ALP), which utilizes the characteristic of the temporal constraint and predicts the most possible RSS value in the next time slot for accurate trace reconstruction over time.

To validate our approach, we conducted experiments in an office building environment across different technologies including IEEE 802.11 b/g and IEEE 802.15.4. We deployed our own traffic sniffers or utilized the existing access points (APs) at fixed locations to collect RSS packets of mobile devices. Our experimental results show that DEMOTE is highly effective in detecting spoofing attacks in mobile environments by using only one AP in the signal space. Further, if the localization process is conducted, DEMOTE can detect spoofing attacks by using the physical position estimates obtained from localization based on multiple APs.

We begin the paper in Section II by putting our work in the broader context of related research. In Section III, we specify the attack model and present the theoretical approach used in the DEMOTE system. We describe our experimental methodology and present the validation results across different wireless technologies in Section IV. Finally, we conclude in Section V.

978-1-4244-2908-0/09/$25.00 ©2009 IEEE

II. Related Work

The traditional security approach to prevent identity fraud is to use cryptographic authentication [7], [8]. As it is not always desirable to use authentication due to limited resources on wireless nodes and infrastructural overhead involved, recently new approaches utilizing wireless transmission properties such as RSS and the wireless channel have been proposed [4], [9]. [4] introduced a security layer that is separate from conventional network authentication methods. They developed forge-resistant relationships based on packet traffic to detect spoofing attacks. [9] utilizes properties of the wireless channel to support security objectives.

To detect mobility of wireless nodes, [10] determined mobility from GSM traces using Euclidean distance in signal space. [11] used RSS collected in wireless LAN to detect wireless device mobility. In [12] signal variance is used with Hidden Markov Model (HMM) to eliminate oscillations between the static and mobile states for mobility detection. Further, [13] proposed to use correlation coefficients on RSS traces to detect wireless devices that are moving together.

The works that are most closely related to us are [3], [5], [6]. [3] proposed the use of matching rules of singalprints such as differential values, max-matches, and min-matches to detect identity-based spoofing. [5] implemented a spoofing detector by utilizing K-means cluster analysis in the signal space. Further, [6] captured the effects of antenna diversity and used Gaussian Mixture Modeling (GMM) for RSS profiling to detect spoofing attacks. Although these methods have varying detection and false alarm rates, none of these approaches can detect spoofing attacks in mobile wireless environments. Our work is novel as it is the first to explore methods for spoofing attack detection when wireless devices are moving around.

III. Detection System Approach

A. Attack Model

In this paper, rather than considering that the victim nodes are static, we focus on the situation when the victim nodes are mobile. We consider the spoofing nodes to be either mobile or static. When both the victim node and the spoofing node are static, spoofing attacks can be detected by using the techniques in previous works [4]–[6] mentioned in Section II. For detecting the mobility of wireless devices, we can use existing metrics, such as the variance of RSS and the techniques presented in [10]–[12]. Thus it is possible to distinguish the mobile nodes from the static nodes in wireless networks.

We deploy traffic observers or use the access points (APs) directly that are at fixed locations to record the Received Signal Strength of packets in the network. When a spoofing attack is conducted, we assume that the victim node, whose identity is cloned by the adversary, is also present in the network. In addition, when the attacker is moving around, we assume that the attacker is not moving together with the victim node, which means that the victim node and the spoofing node have different movement patterns. It is a reasonable assumption because it requires bigger efforts for an attacker to move together with the victim node by tracing the victim node in all the time intervals. In addition, if the spoofing device is co-moving with the victim node, the attacker also increases the possibility of exposing itself to the victim node. We note that under the case that the spoofing attack is present in a different network region of the victim node, a high-level domain management server should be able to detect the attack since the same node identity has appeared in more than one networks.

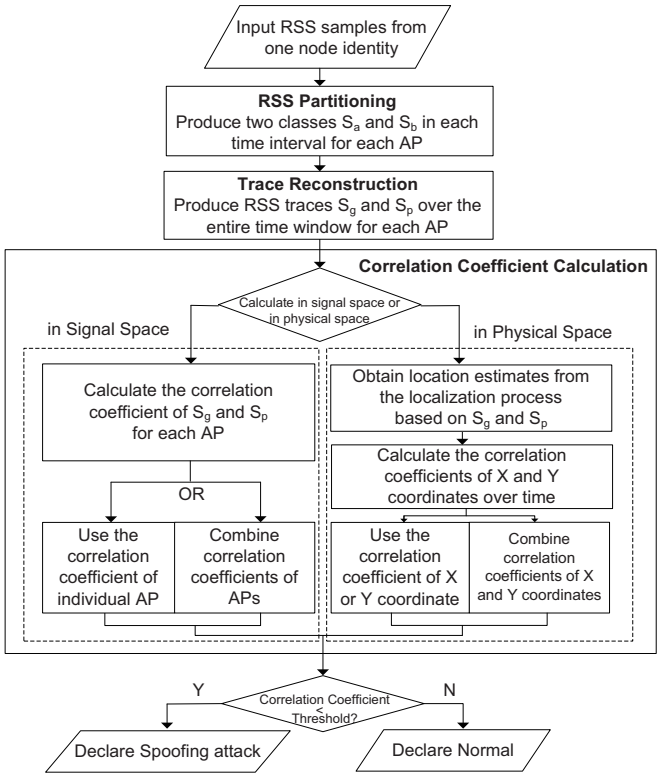


Fig. 1. System flow of DEMOTE.

B. DEMOTE System Overview

DEMOTE performs spoofing attack detection by analyzing the RSS trace for each mobile node identity. RSS is widely available in wireless communication networks and governed by the distance from a device to an AP. This implies that RSS readings are highly correlated with the physical location of a wireless device [5], and thus RSS readings represent a means to distinguish between devices as they move around an environment.

The main idea of the DEMOTE technique is to use the relationship between the RSS and the physical location of a mobile device to perform spoofing attacks detection. If a spoofing attack is present, the RSS trace from *claimed* node identity is the mixture of two RSS traces: one belongs to the victim node and the other belongs to the spoofing node. These two RSS traces are correlated to the different locations of the two physical nodes and are thus not highly correlated to each other. Under normal situations, i.e., there is no spoofing attacks present, the RSS trace from one node identity belongs to one physical node. If the RSS trace is separated into two traces, those two traces are highly correlated to each other as they are determined by the movement pattern of a single mobile node.

To obtain a high detection rate, the key challenge in DEMOTE is to accurately partition and reconstruct the RSS

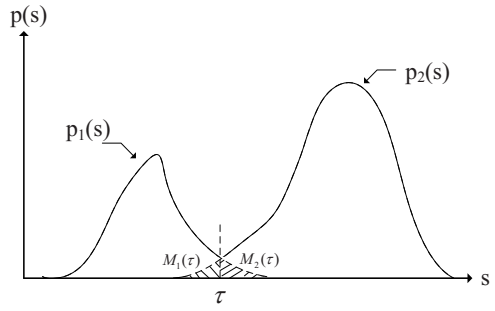


Fig. 2. RSS probability density functions of two classes.

trace that belongs to different physical nodes under a spoofing attack. When the nodes are static, the RSS readings are usually modeled as a Gaussian distribution [14]. The mixed RSS readings from two different nodes can then be modeled as two mixed Gaussian distribution and they can be separated by using the method of Gaussian mixture models [15]. However, when a wireless device is moving around, the distribution of RSS readings is highly dependent on the movement pattern of the node, such as the speed and the direction of the wireless node. Further, it is prohibitive to derive a closed form distribution of RSS trace even with the knowledge of the movement pattern of the wireless node.

Thus, instead of trying to model the RSS readings as any estimated distribution, DEMOTE utilizes an unsupervised thresholding approach to achieve an optimal threshold when performing trace partitioning and separates the RSS readings into two classes. Further, from the two partitioned classes, in order to reconstruct the two RSS traces that belong to two different physical nodes under a spoofing attack or one physical node under a normal situation, we develop the ALignment Prediction (ALP) algorithm that makes use of the temporal constraints inherited from the RSS readings over time, which result from assumptions on the speed and continuity of a mobile device's movement, and helps to predict the most probable RSS value for the next time interval. Finally, DEMOTE computes the correlation coefficients either of the two reconstructed RSS traces in the signal space or of the localization estimates in the physical space based on these two traces. Under a spoofing attack, the two RSS traces come from two different physical devices and thus the value of the correlation coefficient should be low. DEMOTE contains three components: *RSS Partitioning*, *Trace Reconstruction* and *Correlation Coefficient Calculation*. The system flow of DEMOTE is shown in Figure 1. In the following, we describe the theoretic approach of each component in DEMOTE.

C. RSS Partitioning

Suppose the RSS trace from one node identity within time window T of a single AP is $\mathcal{S}$. We equally divide the trace $\mathcal{S}$ into n non overlapping time intervals. Let the RSS readings in the i^{th} time interval be denoted as S_i. Then, $\mathcal{S}$ can be represented as $\{S_1, S_2..., S_n\}$. The objective of the *RSS Partitioning* component is to partition the RSS readings within one time interval, S_i into two classes S_{ai} and S_{bi}, one belongs to the victim node and the other belongs to the spoofing node under a spoofing attack. In this subsection, we first analyze how to obtain the optimal threshold that can minimize the partitioning error of the RSS readings. We then describe a nonparametric and unsupervised method to obtain the optimal threshold for RSS partitioning in DEMOTE.

1) Optimal Thresholding: Let s denote the random variable governing the RSS readings. Suppose the probability density function (PDF) of S_i is $p(s)$. Under a spoofing attack, the density function $p(s)$ is the mixture of two probability densities, one for the victim node and the other for the spoofing node as illustrated in figure 2. Whereas under a normal situation, $p(s)$ is the probability density of a legitimate node. If the form of the densities is known, it is possible to determine an optimal threshold for partitioning the mixture of RSS into two classes in terms of minimum partition error.

Figure 2 shows two probability density functions, the left side one $p_1(s)$ describes the RSS of victim node, while the right side one $p_2(s)$ corresponds to the spoofing node. Then, the mixture PDF describing the overall RSS variation is

$$p(s) = P_{r1}p_1(s) + P_{r2}p_2(s), \tag{1}$$

where P_{r1} and P_{r2} are the probabilities of occurrence of two classes of RSS values respectively. Since any given RSS value either belongs to the victim node or to the spoofing node, we have

$$P_{r1} + P_{r2} = 1. \tag{2}$$

Now, our objective is to select the value of threshold τ that minimizes the average error in making the decisions that a given RSS belongs to the victim node or the spoofing node. Thus, the probability of erroneously classifying a spoofing node's RSS as a victim node's RSS is

$$M_1(\tau) = \int_{-\infty}^{\tau} p_2(s)ds. \tag{3}$$

This is the area under the curve of $p_2(s)$ to the left of the threshold. Similarly, the probability of erroneously classifying a victim node's RSS as a spoofing node's RSS is

$$M_2(\tau) = \int_{\tau}^{+\infty} p_1(s)ds, \tag{4}$$

which is the area under the curve of $p_1(s)$ to the right of τ. Then the overall probability of partition error is

$$M(\tau) = P_{r2}M_1(\tau) + P_{r1}M_2(\tau). \tag{5}$$

Note that the quantities M_1 and M_2 are weighted by the probability of occurrence of the RSS from either the victim node or the spoofing node.

To find the threshold value for which this error is minimal requires differentiating $M(\tau)$ with respect to τ and equating the result to 0:

$$P_{r1}p_1(\tau) - P_{r2}p_2(\tau) = 0. \tag{6}$$

This equation is solved for τ to find the optimum threshold.

Obtaining an analytical expression for τ requires that we know the distributions for the two PDFs which are the $p_1(s)$ and $p_2(s)$ in equation (6) [16]. However, obtaining closed-form solutions of these densities in practice is not feasible when the wireless devices are moving around. Without a priori knowledge on the distribution of each node's RSS distribution, we obtain an optimal threshold by applying the *Otsu* method [17].

2) Trace partitioning approach: We sort RSS values in S_i for the i^{th} time interval. The distinct RSS values are denoted as s_j with $j \in \{1, 2..., L\}$ and L is the number of distinctive values in S_i. Further, the number of RSS samples whose value is s_j is denoted as n_j and the total number of RSS is $N = n_1 + n_2 + ... + n_L$ within the i^{th} time interval. Then the probability distribution of each value can be denoted by

$$p(s_j) = \frac{n_j}{N}, \quad \text{with} \quad p(s_j) \geq 0, \sum_{j=1}^{L} p(s_j) = 1. \tag{7}$$

Under a spoofing attack, s_j can either belong to the RSS class of the victim node or belong to the RSS class of the spoofing node. The values of these two RSS classes usually have overlaps. We use the Otsu method, which uses the discriminate criterion [18] to choose the optimal threshold, in order to make the partitioned classes as tight as possible and thus minimize their overlap. The search criteria for the optimal threshold τ is the minimization of the weighted sum of the variances of two classes (i.e. within-class variance):

$$\delta_w^2(\tau) = P_{r1}(\tau)\delta_1^2(\tau) + P_{r2}(\tau)\delta_2^2(\tau), \tag{8}$$

where $P_{r1}(\tau) = \sum_{j=1}^{(\tau-1)} p(s_j)$, $P_{r2}(\tau) = \sum_{j=\tau}^{L} p(s_j)$ are the probabilities of two classes separated by a threshold τ and $\delta_m^2(\tau)(m = 1, 2)$ is the RSS variance in each class.

To reduce the computational complexity of calculating the within-class variance for each possible threshold when searching for the optimal threshold, we subtract the within-class variance from the variance of the mixture of RSS. We then obtain the between-class variance:

$$\begin{aligned} \delta_b^2(\tau) &= \delta^2 - \delta_w^2(\tau) \\ &= P_{r1}(\tau)[\mu_1(\tau) - \mu]^2 + P_{r2}(\tau)[\mu_2(\tau) - \mu]^2, \end{aligned} \tag{9}$$

where δ^2 is the variance of the mixture of RSS, μ is the mean value of RSS samples in S_i and $\mu_m(m = 1, 2)$ is the mean of each class. Note that $\mu = P_{r1}(\tau)\mu_1(\tau) + P_{r2}(\tau)\mu_2(\tau)$. Substituting μ and simplifying, we get the between-class variance:

$$\delta_b^2(\tau) = P_{r1}(\tau)P_{r2}(\tau)[\mu_1(\tau) - \mu_2(\tau)]^2. \tag{10}$$

Thus, the problem of minimizing the within-class variance is simplified and transferred to maximizing the between-class variance, which utilizes only the zeroth- and the first-order cumulative moments of the RSS value histogram. Further, by using simple recurrence relation we can update the between-class variance as we successively test each threshold:

$$P_{r1}(\tau + 1) = P_{r1}(\tau) + p(\tau) \tag{11}$$

$$P_{r2}(\tau + 1) = P_{r2}(\tau) - p(\tau) \tag{12}$$

$$\mu_1(\tau + 1) = \frac{\mu_1(\tau)P_{r1}(\tau) + p(\tau)\tau}{P_{r1}(\tau + 1)} \tag{13}$$

$$\mu_2(\tau + 1) = \frac{\mu_2(\tau)P_{r2}(\tau) - p(\tau)\tau}{P_{r2}(\tau + 1)} \tag{14}$$

Compared with other unsupervised thresholding methods such as K-means, the Otsu optimal threshold approach is more accurate in when partitioning two classes. Since K-means just measures distances between RSS samples and centroids of classes, while Otsu also takes care of obtaining compact clusters using the inter-class variance [19].

D. Trace Reconstruction

From the RSS Partitioning component, we obtained two RSS classes in each time interval, one belongs to the victim node and the other belongs to the spoofing node under a spoofing attack. For spoofing detection, we further need to reconstruct two RSS traces that are associated with two different nodes respectively over the whole time window T. Thus, the objective of the Trace Reconstruction component is to reconstruct two RSS traces, $S_g = \{s_{gi}\}$ and $S_p = \{s_{pi}\}$ in the time window T such that one trace is associated with the victim node and the other associated with the spoofing node.

Based on the partitioned two classes: S_{ai} and S_{bi} for each time interval, since the time interval is small, we can simply use the average value of RSS of each class in a time interval, represented as $\bar{s}_{ai}$ and $\bar{s}_{bi}$ respectively, for trace reconstruction in T. Then in the i^{th} time interval, trace reconstruction needs to determine whether to assign $\bar{s}_{ai}$ to s_{gi} and $\bar{s}_{bi}$ to s_{pi} or the other way around.

Since we don't have a priori knowledge of the movement patterns of wireless devices, we cannot apply spatial constraint when constructing the RSS trace of a moving node. However, there is a temporal constraint presented in the RSS trace, that is, the RSS samples in the consecutive time intervals are correlated. Thus, although the RSS trace in the whole time window T may not follow any form of curve in practice, the RSS trace within several small time intervals can be modeled to follow a conic curve [20]. In the i^{th} time interval, we can use the RSS values of s_{gi} and s_{pi} to predict the RSS values in the $(i + 1)^{th}$ time interval using conic curve fitting. We then compare the predicted values with $\bar{s}_{a(i+1)}$ and $\bar{s}_{b(i+1)}$, and decide how to assign $\bar{s}_{a(i+1)}$ and $\bar{s}_{b(i+1)}$ to $s_{g(i+1)}$ and $s_{p(i+1)}$.

We developed the ALignment Prediction (ALP) algorithm to predict the RSS values during the trace reconstruction. ALP uses the determined RSS values in the last K time intervals ranging from i^{th} to $(i - K - 1)^{th}$ time intervals to perform conic curve fitting and predict the RSS values, $s^p_{g(i+1)}$ and $s^p_{p(i+1)}$, in the $(i + 1)^{th}$ time interval:

$$s^p_{g(i+1)} = a_{g0i} + a_{g1i}(i + 1) + a_{g2i}(i + 1)^2, \tag{15}$$

and

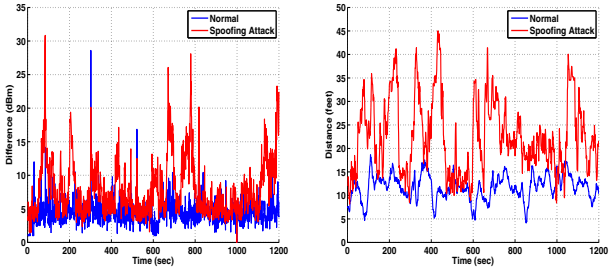
$$s^p_{p(i+1)} = a_{p0i} + a_{p1i}(i + 1) + a_{p2i}(i + 1)^2, \tag{16}$$

where the coefficients $\{a_{g0i}, a_{g1i}, a_{g2i}\}$ and $\{a_{p0i}, a_{p1i}, a_{p2i}\}$ are determined by the latest K values $\{s_{gi}, s_{g(i-1)}, ..., s_{g(i-K-1)}\}$ and $\{s_{pi}, s_{p(i-1)}, ..., s_{p(i-K-1)}\}$ according to the Least-squares polynomial approximation [20]. K is an adjustable variable. In our study, we set $K = 4$. We further define the prediction error as

$$P_{e1} = (s^p_{g(i+1)} - \bar{s}_{a(i+1)})^2 + (s^p_{p(i+1)} - \bar{s}_{b(i+1)})^2, \tag{17}$$

and

$$P_{e2} = (s^p_{g(i+1)} - \bar{s}_{b(i+1)})^2 + (s^p_{p(i+1)} - \bar{s}_{a(i+1)})^2. \tag{18}$$



(a) In Signal Space (b) In Physical Space

Fig. 3. The RSS distance in the signal space and the position distance in the physical space when using the reconstructed RSS traces for spoofing detection.

If $P_{e1} \leq P_{e2}$, we assign $\bar{s}_{a(i+1)}$ to $s_{g(i+1)}$ and $\bar{s}_{b(i+1)}$ to $s_{p(i+1)}$. Otherwise, we assign $\bar{s}_{b(i+1)}$ to $s_{g(i+1)}$ and $\bar{s}_{a(i+1)}$ to $s_{p(i+1)}$.

In the initial setup, when $i = 1$, we set $s_{g1} = \bar{s}_{a1}$, $s_{p1} = \bar{s}_{b1}$ and $s^p_{g2} = s_{g1}$, $s^p_{p2} = s_{p1}$. When $1 < i < K$, we use the first i values of s_{gi} and s_{pi} to fit conic curves and then predict and determine the $(i+1)^{th}$ RSS value.

The pseudo code of the ALP algorithm is shown in Algorithm 1.

Algorithm 1 The ALP algorithm

Input: $\{\bar{s}_{ai}\}$ and $\{\bar{s}_{bi}\}$ with $i = \{1, 2..., n\}$, K;
Output: $\{s_{gi}\}$ and $\{s_{pi}\}$ with $i = \{1, 2..., n\}$;
Initialize: Set $s_{g1} = \bar{s}_{a1}$, $s_{p1} = \bar{s}_{b1}$, $s^p_{g2} = s_{g1}$ and $s^p_{p2} = s_{p1}$;
for $i = 2$ to n **do**
  // Calculate prediction errors P_{e1} and P_{e2};
  $P_{e1} = (s^p_{gi} - \bar{s}_{ai})^2 + (s^p_{pi} - \bar{s}_{bi})^2$;
  $P_{e2} = (s^p_{gi} - \bar{s}_{bi})^2 + (s^p_{pi} - \bar{s}_{ai})^2$;
  // Assign $\bar{s}_{ai}$, $\bar{s}_{bi}$ to s_{gi} and s_{pi} according to prediction errors;
  if $P_{e1} \leq P_{e2}$ **then**
    $s_{gi} = \bar{s}_{ai}$, $s_{pi} = \bar{s}_{bi}$;
  else
    $s_{gi} = \bar{s}_{bi}$, $s_{pi} = \bar{s}_{ai}$;
  end if
  // Calculate coefficients of conic curves;
  if $(i < K)$ **then**
    Obtain $\{a_{g0i}, a_{g1i}, a_{g2i}\}$ and $\{a_{p0i}, a_{p1i}, a_{p2i}\}$ using the first i values of $\{s_{gi}\}$ and $\{s_{pi}\}$;
  else
    Obtain $\{a_{g0i}, a_{g1i}, a_{g2i}\}$ and $\{a_{p0i}, a_{p1i}, a_{p2i}\}$ using the latest K values of $\{s_{gi}\}$ and $\{s_{pi}\}$;
  end if
  // Calculate prediction value $s^p_{g(i+1)}$ and $s^p_{p(i+1)}$ for $(i+1)^{th}$ interval;
  $s^p_{g(i+1)} = a_{g0i} + a_{g1i}(i+1) + a_{g2i}(i+1)^2$;
  $s^p_{p(i+1)} = a_{p0i} + a_{p1i}(i+1) + a_{p2i}(i+1)^2$;
end for

E. Correlation Coefficient Calculation

Once the RSS traces are reconstructed, intuitively we can calculate the distance between two nodes either in the signal space or in the physical space to detect spoofing. However, due to the high variance of RSS that is caused by random noise, environmental bias, and multipath effects [21]. It is not feasible to derive a threshold and distinguish the normal situation from the attack situation most of the time as illustrated in Figure 3. We thus turn to examine the correlation coefficient of the two RSS traces.

Correlation Coefficient. The correlation coefficient measures the degree of linear relationship between two random variables [22]. Instead of calculating the absolute difference of two random variables, the correlation coefficient captures similarities in the changes of two values of random variables. Thus, the correlation coefficient is suitable in determining whether two RSS traces are correlated or not in both signal space and physical space. DEMOTE uses the *Pearson correlation coefficient* [23] to measure the degree of linear relationship between two partitioned traces or their localization results. Given a series of n measurements for random variables X and Y, written as x_i and y_i, where $i = 1, 2, ..., n$, the Pearson correlation coefficient of X and Y is written:

$$r_{xy} = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{(n-1)\delta_x \delta_y}, \tag{19}$$

where $\bar{x}$ and $\bar{y}$ are the sample means of X and Y, δ_x and δ_y are the sample standard deviations of X and Y. The r_{xy} value ranges from -1 to +1. A value of r_{xy} near +1 or -1 indicates a high degree of linearity between X and Y, whereas a value near 0 indicates a lack of such linearity. A positive value indicates that X and Y tend to change together (i.e. decreasing or increasing), whereas a negative value indicates that Y tends to decrease when X increases.

Signal Space. When examining the correlation coefficient in signal space, the random variables X and Y correspond to the constructed RSS traces S_g and S_p. Under normal situations, these two RSS traces should be highly correlated since they are from one mobile wireless node and determined by the same movement pattern of the node. However, under a spoofing attack, these two RSS traces are uncorrelated as they come from two different mobile nodes and determined by the movement pattern of each node separately.

Physical Space. We further study the correlation coefficient in physical space. We can conduct localization [14] to perform location estimation utilizing the two separated RSS traces. The random variables X and Y then correspond to the localization estimates obtained. Under a normal (non-attack) situation, the localization results obtained in physical space are correlated and directly reflect the movement pattern of the wireless node, whereas they are uncorrelated under a spoofing attacks as the movement patterns of the victim node and the spoofing node are different.

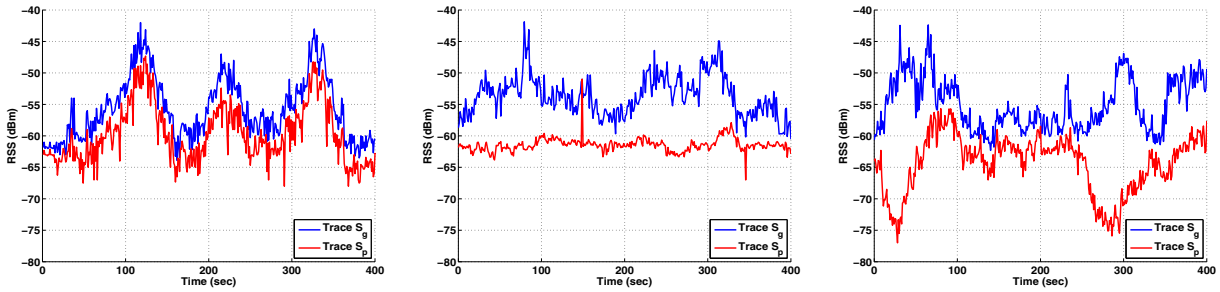
Therefore, by examining the degree of correlation of the RSS traces in either signal space or in physical space, we can determine whether there is spoofing attack present in the network.

IV. Experimental Evaluation

In this section, we first describe our experimental methodology and metrics that we use to evaluate our approach. We then present the experiment results of detecting mobile spoofing attacks.

A. Experimental Methodology

1) Experimental setup: To evaluate the effectiveness of DEMOTE, we conducted experiments using both an 802.11 (WiFi) network as well as an 802.15.4 (ZigBee) network in the Wireless Network Laboratory (WINLAB) at Rutgers University. Figure 4 depicts the layout of experiment site, where the floor size is $219 \times 169 ft$. All experiments were



(a) Normal condition, r_C = 0.92. (b) The spoofing node is static, r_C = 0.1 (c) The spoofing node is mobile, r_C = - 0.19

Fig. 5. The reconstructed RSS traces for access point C under different scenarios in the 802.11 network.

conducted in the yellow shaded area, which is the WINLAB space. We deployed four access points (APs) which were used to observe packet traffic at fixed locations in our experiment. Each access point is shown as a red star in Figure 4 and denoted as A, B, C and D. For the 802.11 (WiFi) network, each access point is a Linux machine with a 1-GHz CPU, 512 MBs of RAM and a 20-GB disk. We used Atheros miniPCI 802.11 wireless card, which connected to an external 7 dBi Omni directional antenna to monitor packet traffic. Whereas for the 802.15.4 (ZigBee) network, we attached a Tmote Sky mote on each access point, which is used in 802.11 (WiFi) network, and each Tmote Sky mote connected to an external 7 dBi Omni directional antenna. We configured each attached mote as an access point to monitor the traffic of the 802.15.4 (ZigBee) network.

To collect RSS traces over time, two people carried one laptop each with a Tmote Sky mote attached to the laptop. During the experiment, these two people either stood or randomly walked around. The experiment was one-hour long and each walking/standing period was ten-minutes long. The walking speed was about 4ft/sec (i.e. normal human walking speed) in the yellow shaded area in Figure 4. The laptops transmitted packets at the rate of 10 packets/sec. Each access point recorded the transmitter's MAC address (for WiFi) or ID (for ZigBee), RSS and timestamp of each packet transmitted from the mobile laptops, and then forwarded to a central server to store. We choose one transmitter as the victim node and the other transmitter as the spoofing attacker. Using this experimental setup, when the victim node is mobile, we can evaluate the effects when the adversary is either static or mobile. Under non-attack situations, the RSS trace from one node identity (i.e. one MAC address or one mote ID) is from one mobile transmitter. Under a spoofing attack, the RSS trace from one node identity is the mixture of two RSS traces from two different transmitters. We choose 20 minutes as the time window T of the RSS trace for both the 802.11 (WiFi) network and the 802.15.4 (ZigBee) network. And the time interval is set to 1 second, thus there are total 1200 intervals (i.e. $n = 1200$) in our experiment.

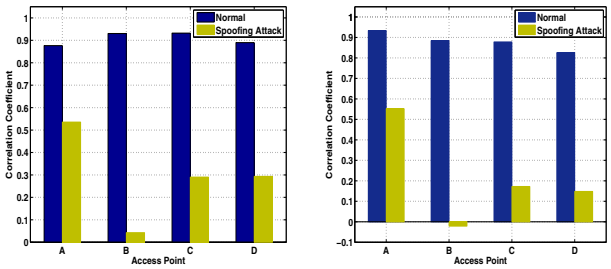
Fig. 4. Layout of the experiment floor and the deployment of access points.

2) Localization Algorithm: We used a scene-matching localization algorithm, called Gridded-RADAR [14], which builds an interpolated radio map to perform localization utilizing the reconstructed RSS traces in each time interval. There are two phases in the algorithm: off line training phase and runtime testing phase. During the off line training phase, a mobile transmitter with known position broadcasts beacons periodically, and the RSS readings are measured at those four access points shown in Figure 4. Collecting together the averaged RSS readings from each of the access point for 101 known locations, shown as small dots in Figure 4, in our experiment provides an interpolated radio map, which serves as training data. During the runtime testing phase, localization is performed by comparing RSS values in the reconstructed RSS traces to the interpolated radio map. The record in the interpolated radio map whose signal strength vector is closest in the Euclidean sense to the observed RSS vector is declared as the location estimation.

3) Metrics: The effectiveness of the detection capability in DEMOTE lies in two aspects: accuracy and efficiency. We will evaluate the detection accuracy in terms of the detection rate and the false positive rate. The detection rate is defined as the percentage of spoofing attack attempts that are determined to be under attack. We declare the presence of a spoofing attack when the computed value of the correlation coefficient is less than a threshold. Further, we define the detection time as the duration of a RSS trace that is needed to calculate the correlation coefficient for spoofing attack detection. The detection time versus the detection rate will be studied to evaluate the efficiency of DEMOTE.

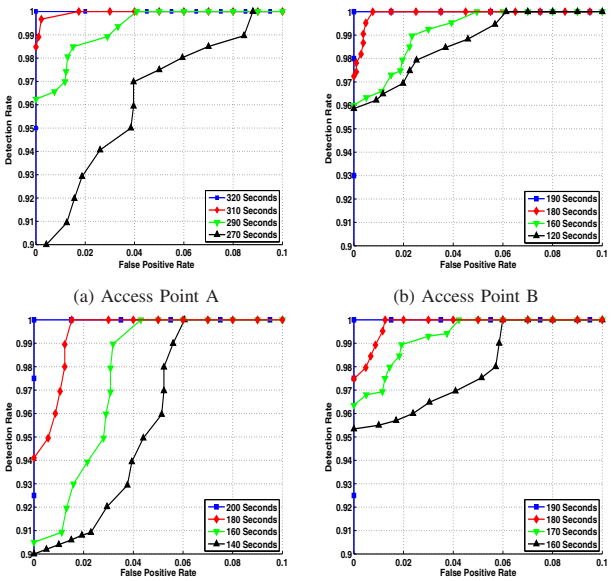
B. Detection in Signal Space

1) Effectiveness: Figure 5 illustrates the reconstructed RSS traces for AP C under three different scenarios: a mobile wireless node under normal situations, under a spoofing attack with a static spoofing node, and under a spoofing attack with a mobile spoofing node. Figure 5(a) shows that under a normal situation the two reconstructed traces are changing together and reflect one movement pattern of the wireless node. We observed a high value of the correlation coefficient of 0.92. Under a spoofing attack, Figure 5(b) and (c) show the RSS



(a) 802.11 network (b) 802.15.4 network

Fig. 6. The correlation coefficient of the reconstructed RSS traces for each access point.



(a) Access Point A (b) Access Point B

(c) Access Point C (d) Access Point D

Fig. 7. ROC curves under different detection time for each AP in the 802.11 network.

traces when the spoofing node is static or moving around respectively. In both figures we were able to extract two distinct RSS traces indicating two different movement patterns, and consequently the corresponding values of the correlation coefficient were low, only 0.1 and -0.19. Since the situation where a spoofing node is static is a very simple case to handle, we focus the rest of our discussion on the harder case where the spoofing node is mobile.

Figure 6 presents the values of the correlation coefficient of the reconstructed RSS traces from each AP for both the 802.11 network and the 802.15.4 network in normal situations and under a spoofing attack respectively. Under non-attack situations, we observed that the values of the correlation coefficient for both networks are consistently high across all access points, above 0.85. Whereas under a spoofing attack, the values of correlation coefficients are much lower. In particular, the correlation coefficients are all below 0.3 for access points B, C and D under a spoofing attack. However, the value of the correlation coefficient of AP A is around 0.5 which is slightly higher, but still much less than those under normal situations. Therefore, in DEMOTE, by choosing an appropriate threshold of the value of the correlation coefficient (e.g., 0.6) we will be able to detect the presence of a spoofing attack effectively.

2) Detection performance using single AP: Figure 7 presents the Receiver Operating Characteristic (ROC) curves under different detection times in the 802.11 network for each AP. We observed that when the detection time is over 160 seconds, the detection rates are above 95% and the false positive rates are below 6% for APs B and D. Further, it takes 190 seconds to achieve 100% detection rate and 0% false positive rate for these two APs. The performance of AP C is slightly worse than the APs B and D. The detection rate is above 90% and the false positive rate is below 6% when the detection time is over 160 seconds. And it takes 200 seconds to achieve 100% detection rate and 0% false positive rate. Thus, the results from APs B, C, and D are encouraging as a high detection rate, over 90%, can be achieved within a short detection time around 160 seconds.

However, for AP A, its performance is worse than other access points in terms of the detection time to achieve a high detection rate. In particular, it takes more than 270 seconds to achieve a detection rate above 90% and 320 seconds in order to achieve 100% detection rate. Additionally, our results of the detection rate and false positive rate versus various detection time for the 802.15.4 network in Figure 8 are similar to that of the 802.11 network. APs B and D have the best performance and AP A has the worst performance. More specifically, it takes 160 seconds for APs B and D to achieve 100% detection rate and 0% false positive rate, whereas AP A needs a longer time, 320 seconds, to achieve the same performance.

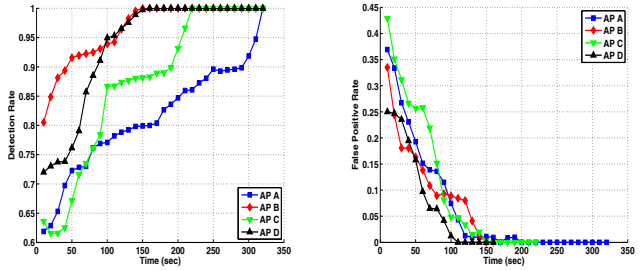
This is inline with our observation of the correlation coefficient for AP A (around 0.5) in Figure 6, which is higher than that from other APs under a spoofing attack. This indicates that bias exists within the RSS traces from each AP when applied to perform spoofing attack detection. Usually, in order to distinguish an unique physical location and consequently determine different moving patterns of nodes in the physical space, we need more than one access point to obtain a distinctive RSS reading in signal space. Using only one access point, we may observe similar RSS traces in signal space even if the moving traces of wireless nodes in the physical space are different.

Therefore, in our experiments, due to the challenges faced when using only one access point, AP A needs to accumulate enough distinctive RSS samples in signal space before it can infer different moving patterns under a spoofing attack. Consequently, we observed that AP A needs the longest time to detect the spoofing attack. An interesting future work item is that when changing the movement patterns of wireless devices, different access points may present longer time to detect an attack.

3) Detection performance using multiple APs: We further study how likely a spoofing attack can be efficiently detected by combing multiple APs. Suppose the value of the correlation coefficient of the reconstructed RSS traces from i^{th} access point is r_i, then the combined correlation coefficient of n access points is

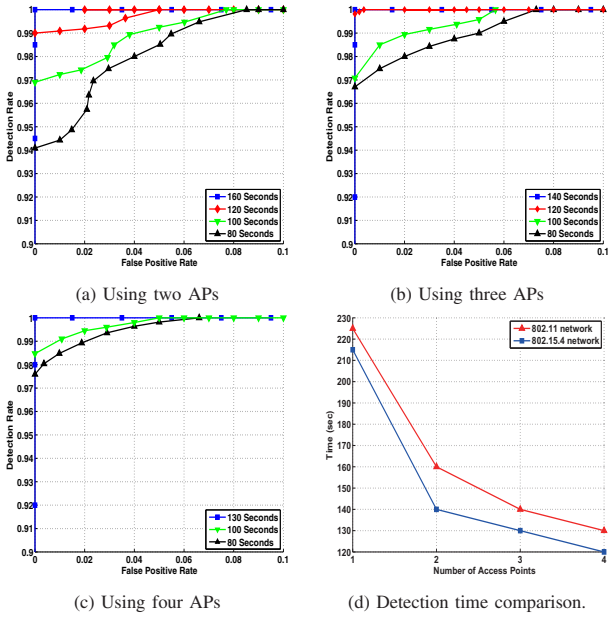
$$r_m = \prod_{i=1}^{n} r_i. \tag{20}$$

We further normalize r_m as needed.



(a) Detection Rate v.s. Detection Time (b) False Positive Rate v.s. Detection Time

Fig. 8. Detection rate and false positive rate as the function of detection time in the 802.15.4 network.



(a) Using two APs (b) Using three APs

(c) Using four APs (d) Detection time comparison.

Fig. 9. (a), (b) and (c) are ROC curves under different detection time when using different number of access points for detection; (d) is the detection time versus different number of APs when achieving 100% detection rate and 0% false positive rate in both networks.

Figure 9(a), (b) and (c) present the ROC curves under different detection time for the 802.11 network when combing 2, 3 and 4 access points, respectively. We observed that the detection rate increases and the detection time decreases when increasing the number of combined access points. Particularly, under the detection time of 80 seconds, the detection rate increases from 94% to 96.8% and further to 97.7% when the number of combined access points increases from 2 to 3 and to 4. Moreover, to achieve 100% detection rate and the 0% false positive rate, the detection time decreases from 160 seconds to 140 seconds and further down to 130 seconds when the number of combined access points increases from 2 to 3 and then to 4.

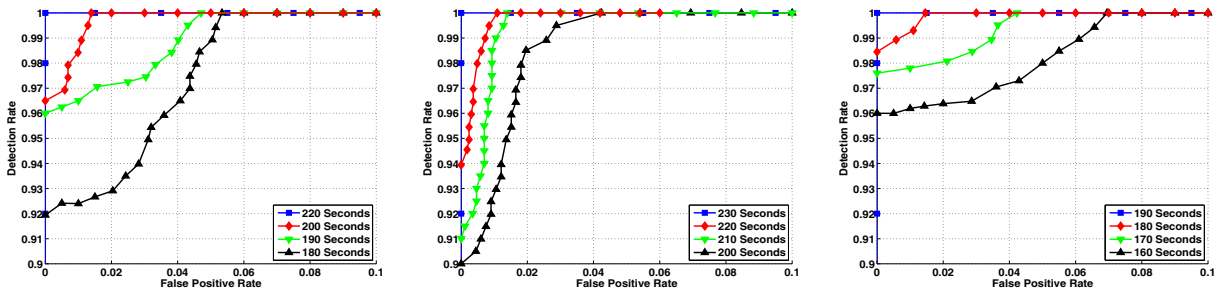
Figure 9(d) presents the comparison of the detection time when combining different number of access points for both the 802.11 network as well as the 802.15.4 network under the detection rate of 100% and the false positive rate of 0%. We observed that the detection time decreases sharply as the number of combined access points increases. Compared to using single AP for attack detection, the key observation is that the detection time is significantly reduced when using multiple APs. Particularly, we observed that the detection time reduced from 225 seconds to 160 seconds in the 802.11 network and from 215 seconds to 140 seconds in the 802.15.4 network. Further, the detection time decreases gradually when further increasing the number of multiple APs from 2 to 4. This is inline with our discussion in Section IV-B2, that is, using a single access point cannot determine the unique location of a wireless device in the physical space, and thus using a single access point takes more time to distinguish the movement patterns for different nodes. Moreover, by using multiple APs the detection time presents a consistent decreasing trend indicating that the bias in RSS traces introduced by individual AP has been smoothed out. Therefore, combining multiple APs for spoofing attack detection helps to achieve a high detection rate and a low false positive rate quickly, which is a critical factor in a spoofing attack detection system such as DEMOTE.

C. Detection in Physical Space

In this section, we perform localization using the reconstructed RSS traces and study the spoofing attack detection capability of DEMOTE in the physical space. Figure 10 presents the ROC curves under different detection time of detecting a spoofing attack in physical space. The location estimation is conducted using the Gridded-RADAR algorithm. Figure 10(a) and (b) show the results by calculating correlation coefficients over the estimated X coordinate and Y coordinate respectively. We found that the results of the X coordinate are similar to those of the Y coordinate. In particular, the detection time that achieves over 90% detection rate with less than 6% false positive rate for the X coordinate is 180 seconds, whereas it is 200 seconds for the Y coordinate. Further, the detection time that achieves 100% detection rate and 0% false positive rate is 220 seconds for the X coordinate, while it's 230 seconds for the Y coordinate.

Figure 10(c) presents the detection results by combining the X and Y coordinates. We use the same method as combining multiple APs to combine the correlation coefficients of X and Y coordinates. We observed that the performance of the combined results is better than the performance using separate X and Y coordinates. This is because our experimental site is a 2D space, it needs both X and Y coordinates to determine an unique location in the physical space. Thus we observed that the detection time is reduced when determining different movement patterns in the physical space by combining X and Y coordinates. Specifically, we found that the detection time can be reduced from 220 seconds to 190 seconds when achieving 100% detection rate and 0% false positive rate and using 160 seconds is enough to detect a spoofing attack of about 96% detection rate and 0% false positive rate.

Compared to the performance in the signal space, the detection performance in the physical space is slightly worse, which is mainly due to the location estimation errors introduced by the localization process. The advantage of conducting spoofing attacks in the physical space is that the localization process can provide additional location information of the victim node as well as the spoofing node, which will further help to infer the movement patterns of these nodes. Once the movement pattern of the spoofing node is traced, one can neutralize the attacker through human intervention.



(a) X coordinate (b) Y coordinate (c) Combining X and Y coordinates

Fig. 10. ROC curves of detecting spoofing attacks in the physical space.

V. Conclusion

As wireless networks are integrated with our daily social lives, there is an increasing need to support emerging mobile wireless applications. One serious class of threats that will affect the successful deployment of mobile wireless applications are spoofing attacks. In this work, we proposed an approach to detect spoofing attacks in mobile wireless environments, which is a problem that has not been addressed in previous work. We developed the DEMOTE system, which utilizes an optimal thresholding scheme to partition the RSS readings and further reconstruct the RSS traces over time for attack detection. Our alignment prediction (ALP) algorithm exploits the temporal constraint in the RSS readings and predicts the best RSS alignment of partitioned RSS classes for RSS trace reconstruction.

To validate the effectiveness of our approach, we conducted experiments using mobile wireless devices across different technologies including IEEE 802.11 b/g and IEEE 802.15.4 in an office building environment. We investigated the detection performance of DEMOTE in terms of detection accuracy and detection efficiency both in the signal space, using either single access points or multiple access points, and in the physical space, using the localization results. Our experimental results provide strong evidence that our system and algorithm is highly effective and efficient in detecting spoofing attacks in mobile environments. Further, we found that under normal (non-attack) situations the reconstructed RSS traces are highly correlated since the traces are originated from one mobile device, whereas under a spoofing attack the RSS traces are much less correlated because of the presence of the spoofing node that is not moving together with the victim node.

Acknowledgments

We gratefully appreciate the help from Gayathri Chandrasekaran, Mesut Ali Ergin, Prof. Marco Gruteser, and Prof. Richard P. Martin in WINLAB who devoted valuable time in collecting the data sets and interpreting the collected data.

References

[1] F. Ferreri, M. Bernaschi, and L. Valcamonici, "Access points vulnerabilities to dos attacks in 802.11 networks," in *Proceedings of the IEEE Wireless Communications and Networking Conference*, 2004.

[2] J. Bellardo and S. Savage, "802.11 denial-of-service attacks: Real vulnerabilities and practical solutions," in *Proceedings of the USENIX Security Symposium*, 2003, pp. 15 – 28.

[3] D. Faria and D. Cheriton, "Detecting identity-based attacks in wireless networks using signalprints," in *Proceedings of the ACM Workshop on Wireless Security (WiSe)*, September 2006.

[4] Q. Li and W. Trappe, "Relationship-based detection of spoofing-related anomalous traffic in ad hoc networks," in *Proceedings of the Third Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks (SECON)*, September 2006.

[5] Y. Chen, W. Trappe, and R. P. Martin, "Detecting and localizing wirelss spoofing attacks," in *Proceedings of the Fourth Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks (SECON)*, May 2007.

[6] Y. Sheng, K. Tan, G. Chen, D. Kotz, and A. Campbell, "Detecting 802.11 MAC layer spoofing using received signal strength," in *Proceedings of the IEEE International Conference on Computer Communications (INFOCOM)*, April 2008.

[7] A. Wool, "Lightweight key management for ieee 802.11 wireless lans with key refresh and host revocation," *ACM/Springer Wireless Networks*, vol. 11, no. 6, pp. 677–686, 2005.

[8] T. Aura, "Cryptographically generated addresses (cga)," *RFC 3972, IETF*, 2005.

[9] Z. Li, W. Xu, R. Miller, and W. Trappe, "Securing wireless systems via lower layer enforcements," in *Proceedings of the ACM Workshop on Wireless Security (WiSe)*, 2006.

[10] T. Sohn, A. Varshavsky, A. LaMarca, M. Y. Chen, T. Choudhury, I. Smith, S. Consolvo, J. Hightower, W. G. Griswold, and E. de Lara, "Mobility detection using everyday gsm traces," in *UbiComp*, September 2006, pp. 212–224.

[11] K. Muthukrishnan, M. Lijding, N. Meratnia, and P. Havinga, "Sensing motion using spectral and spatial analysis of wlan rssi," in *EuroSSC*, October 2007.

[12] J. Krumm and E. Horvitz, "Locadio: inferring motion and location from wi-fi signal strengths," in *MOBIQUITOUS*, Aug 2004, pp. 4–13.

[13] G. Chandrasekaran, M. Ergin, M. Gruteser, R. Martin, J. Yang, and Y. Chen, "Decode: Detecting co-moving wireless devices," in *Proceedings of the Fifth IEEE International Conference on Mobile Ad Hoc and Sensor Systems (MASS)*, 2008.

[14] K. Kleisouris, Y. Chen, J. Yang, and R. P. Martin, "The impact of using multiple antennas on wireless localization," in *Proceedings of the Fifth Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks (SECON)*, June 2008.

[15] R. Redner and H. Walker, "Mixture Densities, Maximum Likelihood and the EM Algorithm," *SIAM Review*, vol. 26, p. 195, 1984.

[16] R. C. Gonzalez and R. E. Woods, *Digital Image Processing*. Prentice Hall, 2007.

[17] N. Otsu, "A threshold selection method from gray-level histograms," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 9, no. 1, pp. 62–66, 1979.

[18] K. Fukunaga, *Introduction to Statistical Pattern Recognition*. Academic Press, 1990.

[19] M. Mancas, B. Gosselin, and B. Macq, "Segmentation using a region-growing thresholding," in *Proc. SPIE*, vol. 5672, 2005, pp. 388–398.

[20] F. Scheid, *Schaum's Outline of Theory and Problems of Numerical Analysis*. McGraw-Hill, 1989.

[21] Y. Chen, J. Francisco, W. Trappe, and R. P. Martin, "A practical approach to landmark deployment for indoor localization," in *Proceedings of the Third Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks (SECON)*, September 2006.

[22] I. Miller and J. Freund, "Probability and statistics for engineers," *PRENTICE-HALL, INC., ENGLEWOOD CLIFFS, NJ 07632(USA), 1984, 530*, 1984.

[23] G. Casella, R. Berger, and R. Berger, *Statistical inference*. Duxbury Press Belmont, Calif, 1990.