Secure RSS-Fingerprint-Based Indoor Positioning: Attacks and Countermeasures

Lizhou Yuan*, Yidan Hu*, Yunzhi Li*, Rui Zhang*, Yanchao Zhang†, and Terri Hedgpeth †
*University of Delaware, Newark, DE, USA
†Arizona State University, Tempe, AZ, USA
{lizhou,yidanhu,liyunzhi,ruizhang}@udel.edu, †{yczhang,terrih}@asu.edu

***Abstract*—Indoor positioning systems (IPS) based on RSS fingerprints have received significant attention in recent years, but they are unfortunately vulnerable to RSS attacks that cannot be thwarted by conventional cryptographic means. In this paper, we identify two practical RSS attacks on RSS-fingerprint-based IPS (RSS-IPS). In both attacks, the attacker learns the RSS-fingerprint database at the IPS server by acting as a normal user repeatedly issuing location queries and then impersonates selected APs with fake ones under his control. By carefully tuning the locations and transmission power of fake APs, the attacker is able to control the RSS experienced by victim users at target locations, leading to either a large location error or the IPS server misled into returning a fake location of the attacker's choice. We further design a fingerprint-matching mechanism based on a novel truncated distance metric as the countermeasure. Trace-driven simulation studies based on real RSS measurement data demonstrate the severe impact of the proposed attacks and also the effectiveness of our countermeasure.**

I. Introduction

Recent years have witnessed the rapid advance in WiFi-based indoor positioning systems (IPS), which have great potential for facilitating indoor human activities. A usable IPS not only can help users navigate in large, unfamiliar indoor venues such as shopping centers, airports, and hospitals, but also can enable numerous location-based services. For example, business owners in shopping malls can offer promotions and targeted advertisements to attract users passing by and thus improve sales. As another example, manufacturers can explore IPS to improve asset tracking and production-flow monitoring. It is projected that WiFi-based indoor location services will generate revenues up to $2.5 billion by 2020 [1].

IPS based on Received Signal Strength (RSS) fingerprints [2], [3] are the most classical IPS built upon the existing indoor WiFi infrastructure. As the name suggests, an RSS-fingerprint-based IPS (RSS-IPS) relies on distinct RSS features as the fingerprints of different indoor locations. A typical RSS-IPS works in two phases. In the offline training phase, the RSS measurements at selected reference locations are collected via either a site survey or mobile crowdsourcing. The collected RSS measurements along with the associated reference locations form a radio map for storage in the IPS server's database. In the online positioning phase, a user who wants to learn his[1] current location submits to the IPS server his RSS measurements, which are compared by the IPS server with stored RSS fingerprints to return the most likely reference position according to certain criteria. RSS-IPS have great potential because they do not require any special hardware or infrastructure update and only explore ubiquitous smartphones and WiFi infrastructures pervasive in large indoor venues where IPS are needed (e.g., shopping centers and airports).

The open nature of wireless medium invites RSS attacks that cannot be thwarted by conventional cryptographic techniques. Early work [4], [5] shows that the attacker can manipulate RSS measurements by placing absorbing materials such as book, water, and foil between transmitting and receiving devices. In addition, recent studies [6], [7] demonstrate that an attacker can easily impersonate legitimate WiFi access points (APs) with off-the-shelf wireless adapters. We observe that by tuning the locations and transmission power of fake APs under his control, the attacker can launch more sophisticated RSS attacks. The impact of such powerful RSS attacks on RSS-IPS and the corresponding countermeasures are still unknown.

In this paper, we introduce two practical RSS attacks against RSS-IPS. In both attacks, the attacker first learns the server-side RSS fingerprints by acting as a normal user to repeatedly submit RSS measurements for learning the corresponding reference positions returned by the IPS server. Armed With learned RSS fingerprints, the attacker then manipulates the RSS from selected APs measured by a target user who wants to find out his location. By doing so, the attacker can either induce a large location error or mislead the server to return an intended fake location with high probability. As a countermeasure, we design a novel RSS-fingerprint-matching mechanism by exploring the redundancy in APs.

Our contributions in this paper are summarized as follows.

- We experimentally validate the feasibility of manipulating RSS measurements at target locations by tuning the locations and transmission power of fake APs. We show that the attacker can manipulate the RSS measurements at target users within 8 dB for 95% of the time.
- We present two novel, practical RSS attacks on RSS-IPS and also propose a countermeasure based on a novel two-side truncated distance metric.
- Trace-driven simulations based on real RSS measurement data confirm both the detrimental impact of our attacks and the efficacy of our countermeasure.

The rest of the paper is structured as follows. Section II briefs the related work. Section III describes a prototype RSS-

[1]No gender implication.

IPS underlying our studies. Section IV presents the proposed attacks and their performance evaluation. Section V illustrates our countermeasure against the identified attacks and evaluates its performance using real measurement data and trace-driven simulations. Section VI concludes this paper.

II. Related Work

We first review some work most germane to our work.

There are many studies involving the impact of RSS attacks on localization mechanisms. Li *et al*. [8] study the impact of multiple physical-layer attacks on several localization algorithms and introduce a median-based distance metric against such attacks. In [9], Chen *et al.* show that RSS attacks can be easily launched by placing absorbing materials (book, water, foil, human body, etc.) between transmitting and receiving devices and analyze the robustness of various localization algorithms to such attacks. In [4], Bauer *et al.* introduce a threshold-detection mechanism for directional RSS attacks. In [5], Li *et al.* study the all-around RSS attack that manipulates the RSS equally at every landmark and introduce a localization mechanism based on relative RSS. A similar idea is also explored in [10], where a differential fingerprint-matching mechanism is introduced to counter the adverse effects of environmental factors. In [11], Yang *et al.* propose a robust localization mechanism based on K-means cluster analysis to cope with corrupted RSS measurements. In addition, a sensor selection mechanism is introduced in [12]. More recently, Li *et al.* [13] study the false data injection attacks in crowdsourced IPSes. Our proposed RSS attacks have not been studied in previous work.

There is also some work on achieving robust RSS-based localization under maximum-likelihood fingerprint matching, e.g., Horus [14]. A robust fingerprint-matching mechanism based on inclusive disjunction is introduced in [15]. In addition, space filtering and sanity check are explored in [16]. Moreover, Laoudias *et al*. [17] propose a likelihood-based fault detection mechanism to ensure localization accuracy when APs fail during the positioning process. Our proposed attacks and countermeasures target IPS like Radar [2], and their extension to Horus-like systems is left as our future work.

Secure localization has been extensively studied in wireless sensor networks (WSNs) [18]. Localization in WSNs is commonly performed by having a few nodes with self-positioning capabilities as anchor nodes to help other sensor nodes obtain locations through triangulation based on time-of-arrival, angle-of-arrival, or other techniques. Examples of secure localization mechanisms for WSNs include location verification based on distance-bounding [19], minimum mean square location estimation [20], etc. This line of research does not use RSS fingerprints and is thus orthogonal to our work.

III. A Prototype RSS-IPS

We built a prototype RSS-IPS based on Radar [2], the most classical RSS-IPS, to illustrate the proposed attacks and countermeasure. The prototype was implemented with Android studio/Java on a Huawei Honor 8 smartphone, which has a 2.3 GHz octa-core CPU, 4 GB RAM, and a 5.2-inch display. The sampling frequency for the WiFi module is 0.67 Hz. We deployed the prototype on a square 17.8m-by-17.8m floor of a university building with the floor plan shown in Fig. 1, where totally $m = 35$ WiFi APs are distributed.

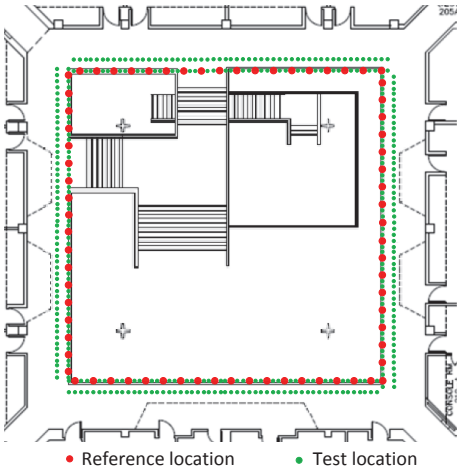


Fig. 1: Indoor floor plan for our experiments.

In the offline phase, we collected the RSS fingerprints at $n = 72$ reference locations, denoted by $x_1, \ldots, x_n$ (red dots in Fig. 1). The RSS fingerprint for each reference position x_i is represented as $\mathbf{rss}_i = (\mathsf{rss}_{i,1}, \cdots, \mathsf{rss}_{i,m})$, where each $\mathsf{rss}_{i,j}$ is the RSS from the corresponding AP j. In the online phase, the prototype system matches received RSS fingerprints based on the nearest neighbor in signal space [2]. In particular, on receiving an RSS fingerprint $\mathbf{rss}_u = (\mathsf{rss}_{u,1}, \cdots, \mathsf{rss}_{u,m})$ from the user, the server returns the reference position x_{i^*} where

$$i^* = \arg\min_{i^* \in [1,n]} \sqrt{\sum_{j=1}^{m} (\mathsf{rss}_{u,j} - \mathsf{rss}_{i,j})^2}. \quad (1)$$

We validated the fidelity of our prototype system by emulating location queries at 360 random positions in the floor plan (i.e., green dots in Fig. 1. We got an average error of 2.01m and a median error of 1.32m, which are quite consistent with the results reported in Radar [2].

IV. Novel RSS Attacks on RSS-IPS

Now we illustrate two novel RSS attacks on RSS-IPS.

A. Overview

We observe that if the attacker knows the server-side RSS fingerprints and can control the RSS experienced by victim users, he can launch more targeted attacks that are much more detrimental than random RSS attacks. In particular, the IPS deployed in many public indoor venues, e.g., shopping malls and airports, are open to the public. By acting as a normal user repeatedly issuing location queries at different locations, the attacker can gradually learn the RSS fingerprints stored at the IPS server. After learning the fingerprint map, the attacker can impersonate selected legitimate APs with fake ones under his control. By tuning the locations and transmission power of

the fake APs, the attacker can control the RSS experienced at target locations or users. The knowledge of the RSS-fingerprint map along with the capability of manipulating RSS measurements allows the attacker to launch more effective, targeted attacks on an RSS-IPS.

In what follows, we first show how the attacker can learn the RSS-fingerprint database with reasonable accuracy. We then experimentally evaluate the degree to which the attacker can control the RSS experienced at a target location by adjusting the position and transmission power of fake APs. Subsequently, we introduce two practical RSS attacks on an RSS-IPS. In the first attack, given a limited number of APs that the attacker can impersonate, he intends to induce the maximum location error for a target user. In the second attack, the attacker aims to mislead the server into returning an intended fake location for a target user with minimum fake APs. Finally, we report the experiment results of the two attacks under different settings.

B. Inferring the RSS-Fingerprint Database

Procedure. The attacker intends to build an RSS-fingerprint database as close to the server-side one as possible. For this purpose, we chose 360 different locations uniformly distributed in the hallway as the attacker's test locations. For each test location, we had one participant emulate the attacker by measuring the RSS of each AP and submitting a location query including RSS measurements to the server, which in turn returns the estimated reference location based on Eq. (1). For each newly received reference location, the attacker creates a new RSS fingerprint in his database. It is possible that multiple location queries would result in the same reference location. In this case, the associated RSS fingerprint is the average of multiple RSS measurements. This exemplary process applies to other scenarios with minimal modification.

Experiment results. Fig. 2 compares the number of inferred reference locations with that of test locations. We can see that as test locations increase from 0 to 400, inferred reference locations increase from 0 to 72. Generally speaking, the attacker learns reference locations quickly at the beginning and can eventually infer all of them. We can expect more efficient ways to infer all reference locations, which are beyond the scope of this paper.

We now measure the similarity between the RSS fingerprint map stored at the server and the one reconstructed by the attacker. Fig. 3 shows the CDF of average RSS difference across 72 reference locations. As we can see, the average difference between server-side RSS measurements and those learned by the attacker over 35 APs is less than 4 dBm and 6 dBm for 50% and 90% of the reference locations, respectively. In addition, Fig. 4 shows the CDF of average RSS difference across 72 reference locations for each AP. It is clear that the RSS difference is less than 5 dBm for 60% of the APs. These results suggest that the attacker can learn the RSS fingerprints stored at the server with reasonable fidelity.

C. Manipulating RSS with Fake APs

In this subsection, we study the feasibility of controlling the RSS experienced by the user at a target location by impersonating legitimate APs.

Procedure. Controlling the RSS at a target location involves two steps. First, the attacker needs to jam legitimate APs' transmission. A recent study [7] has shown that WiFi beacon messages can be easily jammed using USRP or off-the-shelf wireless adapters. Our experiments show that once the attacker identifies the MAC address and beacon interval of the chosen legitimate AP, he can launch either continuous or selective jamming attacks by setting a relatively small beacon interval (e.g., 10.24ms) to overshadow the legitimate beacon frame.

Assuming that the attacker has jammed the chosen legitimate AP's transmissions, his next step is to impersonate the jammed AP with a fake one under his control and then tune the transmission power of the fake AP to control the RSS experienced at the target location. In our experiments, we used the Alfa AWUS036NHA Wireless Adaptor [21] as the fake AP, which is fully configurable and supports a wide range of transmission power levels. We adjusted the transmission power of the fake AP by modifying its driver in Kali Linux.

Experiment results. We conducted two experiments to examine the extent to which the attacker can control the RSS experienced at a target location from a fake AP.

In the first experiment, we placed the fake AP and the receiver in line-of-sight with no barrier in between. We tested 3 different transmission power levels (5dBm, 15dBm, and 30dBm) and 9 distance settings from 2m to 18m. For each transmission power and each distance setting, we collected the RSS measurements for 10 minutes with the sampling rate set to one scan per second, resulting in 600 RSS measurements per experiment configuration. Fig. 5 shows the average RSS measurements varying with transmission power and distance. We can see that the RSS measurements range from -73 dBm to -30 dBm. Generally speaking, the higher the transmission power and the smaller the distance between the fake AP and receiver, the higher the RSS measurement, and vice versa. We did observe some minor fluctuations in RSS measurements, e.g., larger distances with higher RSS measurements, which is mainly due to the rich multi-path effect in the indoor environment. In addition, the RSS measurement is within 5 dBm of its mean for over 95% of the time.

In the second experiment, we covered the fake AP with tinfoil and set its transmission power to 20 dBm. As we can see from Fig. 6, the RSS measurement ranges between -90 dBm and -55 dBm, and it is within 8 dBm of its mean for over 95% of the time. Generally speaking, placing a barrier between the fake AP and the receiver results in lower RSS measurements and larger variance. Combining the results in Figs. 5 and 6, the RSS measurement at the target location ranges between -90 dBm and -30 dBm, which is almost the same as the RSS range a normal user can experience.

While our experiments were not meant to be thorough, the results show that an attacker can control the RSS measurement

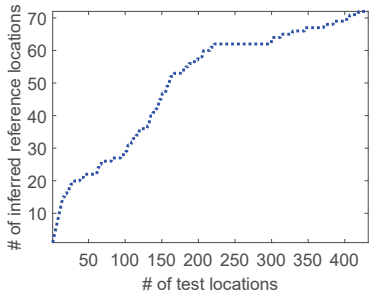


Fig. 2: # of inferred reference locations through location queries.

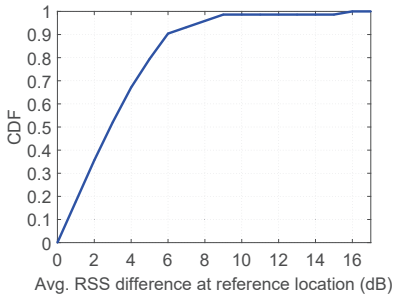


Fig. 3: The CDF of average RSS difference over 35 APs.

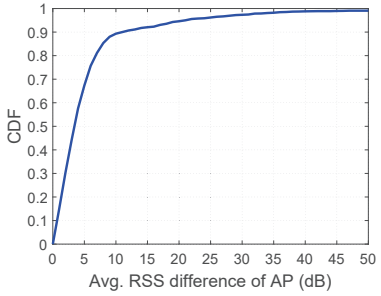


Fig. 4: The CDF of average RSS difference over 72 reference locations.

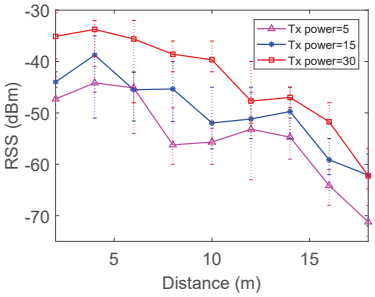


Fig. 5: Impact of distance and Tx power with no barrier.

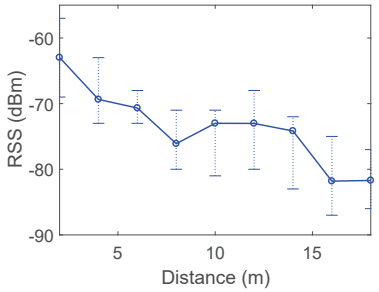


Fig. 6: Impact of distance with a tinfoil barrier.

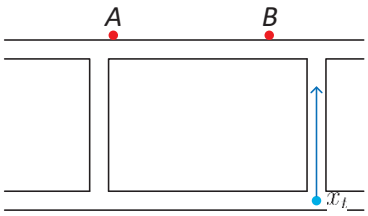


Fig. 7: Motivating example for Attack 1.

at a target location with moderate accuracy by tuning the position and transmission power of fake APs.

D. Attack 1: Maximizing Estimation Distance Error

We now introduce the first RSS attack in which the attacker aims to maximize the location error for users at a target location with limited fake APs under his control.

Motivation. Our first attack is motivated by the following scenario. Considering Fig. 7 as an example, in which two competing stores A and B reside in the same shopping mall that offers indoor positioning and navigation services. Assume that the owner of store A intends to prevent store B's potential customers at location x_t from visiting store B along the direction indicated by the blue arrow. He can do so by manipulating the RSS measurement at location x_t to mislead the server into returning a fake location far away from x_t with limited fake APs under his control.

Problem formulation. We assume that the attacker knows the RSS-fingerprint database $D = \{\langle x_i, \mathbf{rss}_i\rangle\}_{i=1}^n$, where x_i and $\mathbf{rss}_i = (\mathsf{rss}_{i,1}, \cdots, \mathsf{rss}_{i,m})$ denote the reference location and the corresponding RSS fingerprint, respectively. Also assume that the attacker can control the RSS from up to k APs at the target location x_t. The attacker's goal is to manipulate the RSS at x_t such that the server will return a fake reference location $\hat{x}_t$ with the distance between x_t and $\hat{x}_t$ maximized.

A naive solution. A naive way to find the optimal choice of APs for impersonation and their corresponding RSS values is via exhaustive search. Specifically, there are $\binom{n}{k}$ possible ways to choose k APs from totally n APs. Suppose that there are q possible RSS values, where q depends on the range and granularity of RSS measurements. The attacker can try all possible AP and RSS combinations to find the one that leads to the maximum location error via field measurement. The resulting computational complexity is then $\mathcal{O}(\binom{n}{k}q^k)$, which is apparently not scalable to k.

An efficient solution. We now introduce an efficient solution to find the optimal set of APs for impersonation and their corresponding RSS values at location x_t.

The key idea is to first find the set of reference positions that are *feasible* and then determine the one most far away from x_t. Specifically, we call a reference location x_j feasible if by controlling the RSS of up to k APs at location x_t, the server would return x_j. Let x_{t^*} be the target location x_t's closest reference location. The attacker approximates the RSS measurement at x_t by $\mathbf{rss}_{t^*}$ at the reference location x_{t^*}.

We first examine how the distance between two RSS fingerprints can be affected by manipulating the RSS of fake APs. Consider an arbitrary reference location x_j with the RSS fingerprint $\mathbf{rss}_j$. Without manipulating the RSS of any AP at location x_t, the distance between $\mathbf{rss}_{t^*}$ and $\mathbf{rss}_j$ is given by

$$d(\mathbf{rss}_{t^*}, \mathbf{rss}_j) = \sqrt{\sum_{x=1}^{m} (\mathsf{rss}_{t^*,x} - \mathsf{rss}_{j,x})^2}.$$

How close $\mathbf{rss}_{t^*}$ and $\mathbf{rss}_j$ can be after changing k elements among $\mathsf{rss}_{t^*,1}, \cdots, \mathsf{rss}_{t^*,m}$? It is easy to see that the optimal choice is apparently to change the k largest components among $|\mathsf{rss}_{t^*,1} - \mathsf{rss}_{j,1}|, \cdots, |\mathsf{rss}_{t^*,m} - \mathsf{rss}_{j,m}|$ to zero.

Algorithm 1: Min–κ–Dist

Input : Fingerprints $\mathbf{rss}_x$ and $\mathbf{rss}_y$ and the number of APs under control κ

Output: $d_\kappa(\mathbf{rss}_x, \mathbf{rss}_y)$, $\{\langle i, \mathsf{rss}_i\rangle\}_{i\in I}$, and updated $\mathbf{rss}_x$

1 $I \leftarrow \emptyset, P \leftarrow \emptyset, d_\kappa \leftarrow 0$;

2 Let $\pi(1, \ldots, m)$ be the permutation of $(1, \cdots, m)$ *s.t.*

$$|\mathsf{rss}_{x,\pi(1)} - \mathsf{rss}_{y,\pi(1)}| \le \cdots \le |\mathsf{rss}_{x,\pi(m)} - \mathsf{rss}_{y,\pi(m)}|.$$

foreach $i \in \{1, \ldots, m-\kappa\}$ **do**

3 $\quad d_\kappa \leftarrow d_\kappa + (\mathsf{rss}_{x,\pi(i)} - \mathsf{rss}_{y,\pi(i)})^2$;

4 $d_\kappa(\mathbf{rss}_x, \mathbf{rss}_y) \leftarrow \sqrt{d_\kappa}$;

5 **foreach** $i \in \{\kappa+1, \ldots, m\}$ **do**

6 $\quad P \leftarrow P \bigcup \{\langle \pi(i), \mathsf{rss}_{y,\pi(i)}\rangle\}$;

7 $\quad \mathsf{rss}_{x,\pi(i)} = \mathsf{rss}_{y,\pi(i)}$;

8 **return** $d_\kappa(\mathbf{rss}_x, \mathbf{rss}_y), P, \mathbf{rss}_x$;

Algorithm 2: Maximize location error

Input : Fingerprint database $\{x_i, \mathbf{rss}_i\}_{i=1}^n$, reference location x_{t^*}, and k

Output: Reference location $x_{\max}$, maximal distance $d_{\max}$, and $\{\langle i, \mathsf{rss}_i\rangle\}_{i\in I}$

1 $x_{\max} \leftarrow x_{t^*}, d_{\max} \leftarrow 0, P \leftarrow \emptyset$;

2 **foreach** $j \in \{1, \ldots, n\}$ **do**

3 $\quad$ flag $\leftarrow$ true;

4 $\quad \langle d_\kappa(t^*, j), Q, \mathbf{rss}_t^*\rangle \leftarrow$ Min–κ–Dist$(\mathbf{rss}_t^*, \mathbf{rss}_j, m-k)$;

5 $\quad$ **foreach** $y \in \{1, \ldots, m\} \setminus \{j\}$ **do**

6 $\quad\quad$ **if** $d_\kappa(t^*, j) < d(\mathbf{rss}_j, \mathbf{rss}_y)$ **then**

7 $\quad\quad\quad$ flag $\leftarrow$ false;

8 $\quad\quad\quad$ **break**;

9 $\quad$ **if** flag $=$ true **and** $d_\kappa(t^*, j) > d_{\max}$ **then**

10 $\quad\quad x_{\max} \leftarrow j, d_{\max} \leftarrow d_\kappa(t^*, j), P \leftarrow Q$;

11 **return** $x_{\max}, d_{\max}, P$;

We define the *minimal-κ-dimension distance* between $\mathbf{rss}_i$ and $\mathbf{rss}_j$ as the minimum Euclidian distance between them that can be achieved by changing κ elements in $\mathbf{rss}_i$. The minimal-κ-dimension distance between any two RSS fingerprints can be computed efficiently using Algorithm 1, which takes $\mathbf{rss}_x$ and $\mathbf{rss}_y$ and the number κ of fake APs as input and outputs the minimal-κ-dimension distance along with the set of APs and their corresponding RSS. The complexity of Algorithm 1 is $\mathcal{O}(m \log m)$, as the complexity of the sorting step in Line 2 dominates other operations.

Note that here we abuse the term "distance", as $d_\kappa(\cdot, \cdot)$ does not satisfy the triangle inequality and is thus not a distance metric in the strict sense. For example, since $d_1((0,0),(2,2)) = 2$, $d_1((0,0),(2,0)) = 0$, and $d_1((2,0),(2,2)) = 0$, we have $d_1((0,0),(2,2)) > d_1((0,0),(2,0)) + d_1((2,0),(2,2))$, an example of violating the triangle inequality.

We now discuss the necessary conditions for a reference location to be feasible. Consider again location x_j with RSS fingerprint $\mathbf{rss}_j$ and location x_{t^*} with RSS fingerprint $\mathbf{rss}_{t^*}$. After manipulating the RSS of k APs at location x_{t^*} to achieve the *minimal-κ-dimension distance*, the user's RSS measurement at location x_t changes from $\mathbf{rss}_{t^*}$ to $\mathbf{rss}'_{t^*} = (\mathsf{rss}'_{t^*,1}, \cdots, \mathsf{rss}'_{t^*,m})$, where

$$\mathsf{rss}'_{t^*,x} = \begin{cases} \mathsf{rss}_{t^*,x} & \text{if } x \notin \{\pi(m-k), \cdots, \pi(k)\}, \\ \mathsf{rss}_{j,x} & \text{otherwise.} \end{cases} \tag{2}$$

If the user submits RSS measurement $\mathbf{rss}'_{t^*}$ to the server, the server will return reference location x_j if and only if $\mathbf{rss}'_{t^*}$ is the nearest neighbor of $\mathbf{rss}_j$ in the signal space. In other words, reference location x_j is feasible if and only if $d_k(\mathbf{rss}'_{t^*}, \mathbf{rss}_j) < d(\mathbf{rss}_x, \mathbf{rss}_j)$ for all $x \in \{1, \cdots, n\} \setminus \{t^*\}$.

After finding the set of all feasible locations, the attacker can find the one with the largest distance from the target user's true location x_t. We summarize the whole process in Algorithm 2, which takes the fingerprint database $\{x_i, \mathbf{rss}_i\}_{i=1}^n$, the reference location x_{t^*}, and the number of fake APs as input. Algorithm 2 outputs the feasible reference location $x_{\max}$ with the maximum location error $d_{\max}$ along with the set of APs for impersonation and their corresponding RSS values. The complexity of Algorithm 2 is $\mathcal{O}(nm^2)$, as the complexity of Line 6 dominates other operations.

E. Attack 2: Targeted Location Manipulation

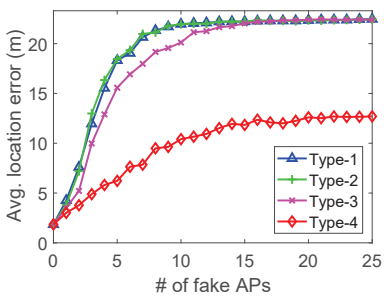
In the second RSS attack, the attacker seeks to mislead the server into returning an intended fake location. Specifically, for a target user at the true location x_t, the attacker aims to let the server return a fake reference location x_c of his choice upon receiving the RSS fingerprint from the target user.

We would like to answer the following questions. For any given pair of locations x_t and x_c, what is the minimal number of APs the attacker needs to impersonate in order to succeed? Which APs does the attacker need to impersonate? What are their corresponding RSS at location x_t? These questions can be answered easily based on the insight from Attack 1. In particular, the minimal number of APs needed by the attacker can be determined by checking whether location x_c is feasible with k fake APs for different ks and then finding the smallest k. We summarize the procedure in Algorithm 3, which takes the fingerprint database, the target user's true location x_u, and the intended fake location x_c as input. With computational complexity $\mathcal{O}(nm^2)$, Algorithm 3 eventually outputs the minimal set of APs for the attacker to impersonate along with their RSS values. In the attack phase, the attacker only needs to find the smallest k with computational complexity $\mathcal{O}(n)$.

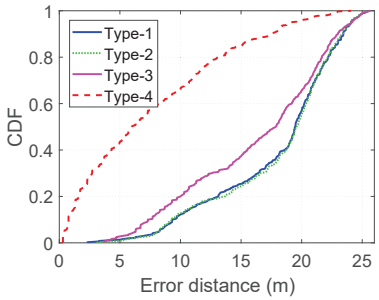
If the number of fake APs needed for the attacker to succeed are beyond his capability, he may alternatively seek to find the optimal set of APs and corresponding RSS such that the location estimated by the server is as close to his intended location x_c as possible. This can be done by finding all the feasible locations and then choosing the one that is closest to x_c. The corresponding computational complexity is $\mathcal{O}(kn)$.

F. Attack Evaluation

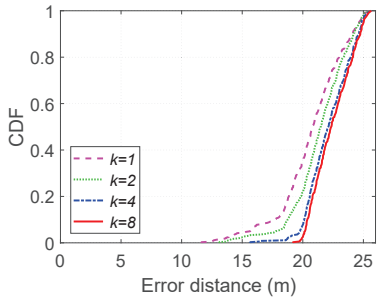
We now evaluate the performance of the two attacks using a combination of experiments and trace-driven simulations.



(a) Average location error

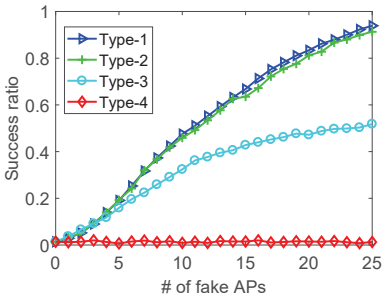


(b) CDF of average location error for $k = 6$

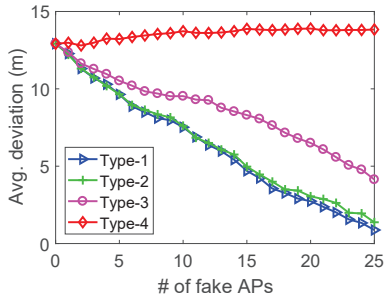


(c) Average location error of Type-3 attackers

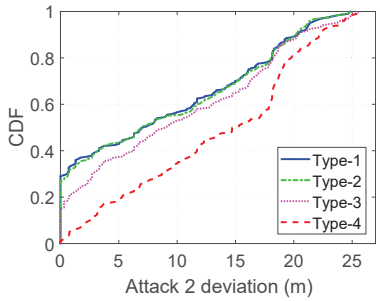
Fig. 9: Experiment results of Attack 1.



(a) Success ratio of four types of attackers



(b) Distance deviation for various attackers



(c) Distance deviation of Type-3 attackers $k = 8$

Fig. 10: Experiment results of Attack 2.

Algorithm 3: Minimal Fake APs for Attack 2

Input : Fingerprint database $\{x_i, \mathbf{rss}_i\}_{i=1}^n$, target user location x_u, and chosen reference location x_c

Output: The minimal number of APs needed and their RSSs for Attack 2

$k \leftarrow 0, P \leftarrow \emptyset$;
foreach $j \in \{1, \ldots, m\}$ **do**
  flag $\leftarrow$ true;
  $\langle d_\kappa(u,c), Q, \mathbf{rss}_u^* \rangle \leftarrow$ Min–κ–Dist$(\mathbf{rss}_u, \mathbf{rss}_c, j)$;
  foreach $y \in \{1, \ldots, n\} \setminus \{u, c\}$ **do**
    if $d_\kappa(u,c) > d(\mathbf{rss}_u^*, \mathbf{rss}_y)$ **then**
      flag $\leftarrow$ false;
      break;
  if flag $=$ true **then**
    $k \leftarrow y, P \leftarrow Q$;
    break;
return k, P;

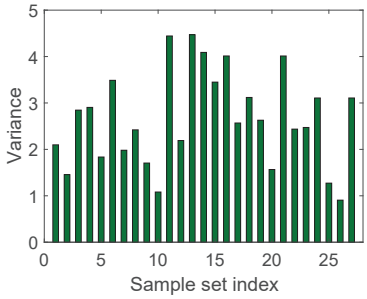


Fig. 8: Variance of RSS distribution in 27 sample sets.

1) Experiment setup: To emulate the attacker's capability of controlling the RSS of the fake AP at the target location, we fit each of the 27 sets of RSS samples in Section IV-C to a normal distribution. Fig. 8 shows the variance of the 27 Gaussian distributions. The average variance across 27 Gaussian distributions is 2.65 dB. For Attack 1, we randomly chose a target location in the indoor floor plan, for which 100 RSS measurements were generated. For Attack 2, we randomly generated 1000 pairs of target and intended fake locations. We consider the following four types of attackers.

- **Type-1 attacker: perfect knowledge of RSS fingerprints and perfect control of fake RSS**. The attacker knows the exact server-side fingerprint database and can precisely control fake APs' RSS experienced by victim users. The performance under this unlikely ideal case serves as a baseline for comparison purposes.
- **Type-2 attacker: perfect knowledge of RSS fingerprints and imperfect control of fake RSS**. The attacker knows the exact server-side fingerprint database (e.g., by compromising the IPS server) but can only control fake APs' mean RSS experienced by victim users. In particular, the fake RSS measurement at the victim follows the Gaussian distribution $\mathcal{N}(\mu, \sigma)$, where μ is the attacker's intended mean RSS value computed from Algorithm 1 or 2, and $\sigma = 2.65$ dBm.
- **Type-3 attacker: imperfect knowledge of RSS fingerprints and imperfect control of fake RSS**. The attacker learns an approximate copy of the server-side fingerprint

database through the procedure in Section IV-B, and he can only control the mean fake RSS experienced by the user just as what a Type-2 attacker does.

- **Type-4 attacker: random RSS attacks**. The attacker has no knowledge about the server-side fingerprint database. He also randomly chooses legitimate APs to impersonate with random transmission power, and the RSS experienced by the user follows the uniform distribution $\mathcal{U}(-95\text{dBm}, -35\text{dBm})$.

2) Performance evaluation of Attack 1: Fig. 9(a) compares the average location error the four types of attackers can each achieve with fake APs varying from 0 to 25. We can see that the average location error increases with fake APs in all four cases, which is expected. Among them, the Type-1 attacker is obviously most detrimental due to his most powerful capability. In addition, the average location errors of Type-2 and Type-3 attackes are both very close to that of the Type-1 attacker; this result indicates that the imperfect control of fake RSS and the imperfect knowledge of the RSS-fingerprint database both have very limited impact on the resulting location error. Finally, the Type-4 attacker's average location error always fall between 1/3 to 1/2 of that of the other three types under the same conditions. For example, with six fake APs, the average location errors of Type-1 and Type-4 attackers are 15.2m and 6.1m, respectively. These results show that Attack 1 is highly effective under practical conditions.

Fig. 9(b) shows the CDFs of the location error that the four types of attackers can achieve with six fake APs. Similar to what we have observed from Fig. 9(a), Type-1, Type-2, and Type-3 attackers have very close performance and all significantly outperform the Type-4 attacker. For example, the median location error achievable by Type-3 and Type-4 attackers are over 17m and below 6m, respectively. This result further confirms that Attack 1 is much more effective than random RSS attacks in practical scenarios.

Fig. 9(c) shows the CDFs of the location error under different number k of fake APs controlled by Type-3 attacker. We can see that the location error increases as k grows, which is expected. In addition, the median location error is over 18 meters under all ks, which is approximately the length of the hallway in the experimental floor. These results indicate even a smaller number of fake APs can greatly degrade the positioning accuracy.

3) Performance evaluation of Attack 2: As in Section IV-F2, we tested 1000 pairs of target and intended locations for all four types of attackers. We use the following two metrics to evaluate the performance of Attack 2.

- *Success ratio*: the ratio at which the attacker can successfully mislead the server to return an intended location.
- *Average distance deviation*: the average distance between the attacker's intended location and the closest reference location the server returns under the attacker's influence. The deviation distance is zero for a successful attack.

Fig. 10(a) shows the success ratios of four attacker types when the number k of fake APs changes. We can see that the success ratio increases as k increases for Type-1 and 2 attackers. This is expected, as the more APs the attacker can control, the more likely the server can be cheated into returning the attacker's intended location under Attack 2. In addition, the success ratio of the Type-3 attacker increases as k goes from 0 to 10 and then becomes relatively stable as k further increases. The initial increase is due to the same reason for that of Type-1 and Type-2 attackers. However, since the Type-3 attacker only knows an approximate copy of the server-side RSS-fingerprint database, the difference between the learned RSS fingerprint and the actual RSS fingerprint constrains his success ratio. Finally, the success ratio of the Type-4 attacker is always close to zero and not affected by the change in k. This is also anticipated because random RSS attacks can hardly lead to predictable results. Generally speaking, though the Type-3 attacker is not as effective as Type-1 and 2 attackers, it still has significant advantages over random RSS attacks.

Fig. 10(b) compares the distance deviations of the four attacker types for a varying number k of fake APs. Similar to what we have observed in Fig. 10(a), the distance deviation of the Type-4 attacker is the highest and relatively insensitive to the change in k. In contrast, the distance deviations of Type-1, 2 and 3 attackers all decrease as k increases. Among them, the Type-1 attacker has the smallest distance deviation, which is also expected due to his most powerful capability.

Fig. 10(c) compares the CDFs of the distance deviations for $k = 8$. We can see that the distance deviations of Type-1, 2, and 3 attackers are zero for more than 31.8%, 30.2%, and 18.3% of the time, respectively. These results are in sharp contrast to that of the Type-4 attacker, 0.8% of the time.

V. Countermeasure

In this section, we introduce a novel fingerprint-matching mechanism against the identified RSS attacks.

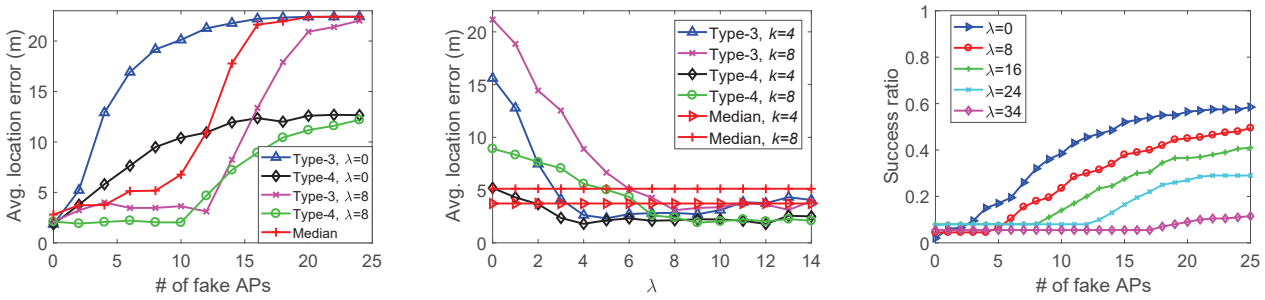
A. Truncated-Distance-Based Fingerprint Matching

We propose a novel truncated-distance-based fingerprint matching (TDFM) mechanism against Attack 1, Attack 2 and random RSS attacks on an RSS-IPS. Assume that the attacker can impersonate k APs. Under Attack 1 and Attack 2 introduced in Section IV, k elements among $|\mathsf{rss}_{t^*,1} - \mathsf{rss}_{j,1}|, \cdots, |\mathsf{rss}_{t^*,m} - \mathsf{rss}_{j,m}|$ would approach zero. If the attacker launches random RSS attacks instead, we expect that k elements among $|\mathsf{rss}_{t^*,1} - \mathsf{rss}_{j,1}|, \cdots, |\mathsf{rss}_{t^*,m} - \mathsf{rss}_{j,m}|$ are likely to be relatively large. To simultaneously defend against Attack 1, Attack 2 and random RSS attacks, we need eliminate the impact of both the smallest k and the largest k elements on the fingerprint-matching result.

Based on the above idea, we define the two-side λ-truncated distance between two RSS fingerprints as follow.

Definition 1. *(***Two-side λ-truncated distance***) Let λ be an even number. For any two RSS fingerprints $\mathbf{rss}_x$ and $\mathbf{rss}_y$, let $\pi(1, \cdots, m)$ be the permutation of $(1, \cdots, m)$ such that*

$$|\mathsf{rss}_{x,\pi(1)} - \mathsf{rss}_{y,\pi(1)}| \leq \cdots \leq |\mathsf{rss}_{x,\pi(m)} - \mathsf{rss}_{y,\pi(m)}|.$$



(a) Attack 1 with TDFM (b) Attack 1 with TDFM (c) Attack 2 with TDFM

Fig. 11: Experiment results of TDFM.

The two-side λ-truncated distance between two RSS fingerprints $\mathbf{rss}_x$ and $\mathbf{rss}_y$ is given by

$$d_{\lambda^+_-}(\mathbf{rss}_x, \mathbf{rss}_y) = \sqrt{\sum_{i=\lambda/2+1}^{m-\lambda/2} (\mathsf{rss}_{x,i} - \mathsf{rss}_{y,i})^2}.$$

Similar to the minimal-κ-dimension distance defined in Section IV-D, we here abuse the notion of "distance", as $d_k^2(\cdot,\cdot)$ does not satisfy the triangle inequality and thus is not a distance metric in the strict sense.

Truncated-distance-based fingerprint matching is similar to Radar [2] except for the distance metric. In particular, in the online phase, the IPS server matches received RSS fingerprints based on the two-side λ-truncated distance. On receiving an RSS fingerprint $\mathbf{rss}_u = (\mathsf{rss}_{u,1}, \cdots, \mathsf{rss}_{u,m})$ from the user, the server returns the reference position x_{i^*} where

$$i^* = \arg\min_{i^* \in [1,n]} d_{\lambda^+_-}(\mathbf{rss}_u, \mathbf{rss}_i). \tag{3}$$

Here λ is a system parameter representing the tradeoff between attack resilience and positioning accuracy without any attack. Generally speaking, the larger λ, the higher attach resilience, the lower positioning accuracy, and vice versa. Its impact is evaluated in the next subsection. It is not difficult to see that the proposed TDFM mechanism is a generalization of Radar [2] based on Euclidian distance and the mechanism proposed in [8] based on median distance. In particular, two-side λ-truncated distance is equivalent to Euclidian distance when $\lambda = 0$ and median distance when $\lambda = (m-1)/2$.

B. Experiment Results

We now report the experiment results of TDFM. We focus on Type-3 and Type-4 attackers who require much fewer resources and thus are more practical than Type-1 and Type-2 attackers. In addition, we compare TDFM with the mechanism based on median distance proposed in [8].

1) TDFM performance under Attack 1: Fig. 11(a) compares the average location errors of TDFM and the median-based mechanism [8] for Type-3 and Type-4 attackers as the number k of fake APs varing from 0 to 25. We can see that the average location error increases with k for the median-based mechanism. In contrast, the average location errors of Type-3 and Type-4 attackers under TDFM with $\lambda = 8$ are relatively stable as k increases from 0 to 10 and then both increase sharply as k further increases. The reason is that by dropping the $\lambda/2$ smallest and $\lambda/2$ largest distance components, neither Attack 1 (Type-3 attackers) nor random RSS attacks (Type-4 attackers) can significantly change the two-side λ-truncated distance between the original RSS measurement and the fake RSS fingerprint when the number of fake APs is not much larger than λ. In addition, the average location errors of Type-3 and Type-4 attackers with $\lambda = 8$ are both consistently smaller than that of the median-based mechanism, so TDFM is more resilient than the median-based mechanism to Attack 1.

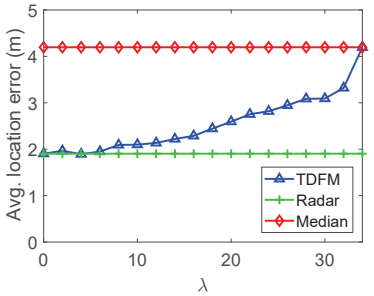


Fig. 12: TDFM under no attack.

Fig. 11(b) compares the average location errors of TDFM for Type-3 and Type-4 attackers with λ varying from 0 to 14. Since the median-based mechanism is not affected by λ, its average location error is plotted for reference only. As we can see, the average location errors of Type-3 and Type-4 attackers are much higher than that of the median-base mechanism when λ is small. The reason is that when the number of fake APs is larger than λ, the attacker is always able to affect some distance components. In contrast, the median-based mechanism matches RSS fingerprints using only one distance component, and the matching result is not affected by fake APs when they are minority. In addition, the average location error of TDFM decreases as λ increases and drops below that of the median-based mechanism when λ surpasses k. So TDFM is more resilient than the median-based mechanism when λ is larger than the number of fake APs.

2) TDFM performance under Attack 2: Fig. 11(c) compares the success ratios of Type-3 attackers under Radar (corre-

sponding to $\lambda = 0$), the median-based mechanism (i.e., corresponding to $\lambda = 34$), and TDFM with k varying from 0 to 25. As we can see, the median-based mechanism has the lowest success ratio always close to zero when k is less than 17. The reason is that by impersonating less than half of the APs, the attacker is unable to affect the median distance between two RSS fingerprints, rendering the RSS attack ineffective. A similar trend holds for TDFM. In particular, the success ratio with TDFM is close to zero if k is less than $\lambda/2$ and increases as k further increases. It is of no surprise that Attack 2 has the highest success ratio under Radar, as the attacker is easy to manipulate the Euclidian distance between two fingerprints. Generally speaking, the higher λ, the higher resilience to Attack 2, and vice versa.

3) TDFM performance under no attack: We also compare the positioning accuracy for Radar, the median-based approach, and TDFM when there is no attack. As we can see from Fig. 12, the median-based mechanism has the largest average distance error, while Radar has the smallest. The average distance error with TDFM increases from that of Radar as λ increases from 0 and then reaches that of the median-based mechanism when λ reaches 34. This result is expected, as the more RSS elements are considered in the distance metric, the higher the positioning accuracy when there is no attack, and vice versa. Moreover, the average distance error of TDFM is quite close to that of Radar for moderate λ.

4) Summary of Experiment Results: We summarize the experiment results as follows.

- TDFM is much more resilient to Attack 1 and random RSS attacks than the median-based mechanism under practical conditions.
- TDFM is less resilient to Attack 2 than the median-based mechanism because it considers more RSS elements during positioning.
- With moderate λ (e.g., approximately half APs are impersonated), TDFM can achieve similar resilience to Attack 2 but with much higher positioning accuracy than the median-based mechanism.

VI. Conclusion

In this paper, we demonstrated two novel RSS attacks on Indoor Position Systems (IPS) based on RSS fingerprints. Armed with the (im)perfect knowledge of the server-side RSS-fingerprint database and the capability of manipulating the RSS experienced at a target location, the attacker can either maximize the location error or mislead the IPS server into returning an intended wrong location. We also proposed a countermeasure based on a novel truncated distance metric. Trace-driven simulation studies based on real RSS measurement data confirmed the severe impact of the proposed attacks and also validated the effectiveness of our countermeasure.

Acknowledgment

We would like to thank the anonymous reviewers for their insightful comments that help improve the quality of this paper. This work was supported in part by the US National Science Foundation under grants CNS-1700032, CNS-1700039, CNS1651954 (CAREER), CNS-1718078, CNS-1514381, CNS-1619251, CNS1421999, and CNS-1320906.

References

[1] "Wi-fi indoor location in retail worth $2.5 billion by 2020." [Online]. Available: https://www.abiresearch.com/press/wi-fi-indoor-location-retail-worth-25-billion-2020

[2] P. Bahl and V. N. Padmanabhan, "Radar: an in-building rf-based user location and tracking system," in *IEEE INFOCOM'00*, Tel Aviv, Israel, March 2000, pp. 775–784.

[3] P. Bahl, V. N. Padmanabhan, and A. Balachandran, "Enhancements to the radar user location and tracking system," *Microsoft Research*, vol. 2, no. MSR-TR-2000-12, pp. 775–784, Feb. 2000.

[4] K. Bauer, D. McCoy, E. Anderson, M. Breitenbach, G. Grudic, D. Grunwald, and D. Sicker, "The directional attack on wireless localization -or- how to spoof your location with a tin can," in *GLOBECOM'09*, Honolulu, HI, Nov 2009, pp. 1–6.

[5] X. Li, Y. Chen, J. Yang, and X. Zheng, "Designing localization algorithms robust to signal strength attacks," in *IEEE INFOCOM'11*, Shanghai, China, April 2011, pp. 341–345.

[6] N. O. Tippenhauer, K. B. Rasmussen, C. Pöpper, and S. Čapkun, "Attacks on public wlan-based positioning systems," in *ACM MobiSys'09*, Krakow, Poland, June 2009, pp. 29–40.

[7] M. Vanhoef and F. Piessens, "Advanced wi-fi attacks using commodity hardware," in *ACSAC'14*, New Orleans, Louisiana, 2014, pp. 256–265.

[8] Z. Li, W. Trappe, Y. Zhang, and B. Nath, "Robust statistical methods for securing wireless localization in sensor networks," in *IPSN'05*, Boise, ID, April 2005, pp. 91–98.

[9] Y. Chen, K. Kleisouris, X. Li, W. Trappe, and R. P. Martin, "A security and robustness performance analysis of localization algorithms to signal strength attacks," *ACM Trans. Sen. Netw.*, vol. 5, no. 1, pp. 2:1–2:37, Feb. 2009.

[10] N. Chang, R. Rashidzadeh, and M. Ahmadi, "Robust indoor positioning using differential wi-fi access points," *IEEE Transactions on Consumer Electronics*, vol. 56, no. 3, pp. 1860–1867, Aug 2010.

[11] J. Yang, Y. Chen, V. B. Lawrence, and V. Swaminathan, "Robust wireless localization to attacks on access points," in *IEEE Sarnoff Symposium'09*, March 2009, pp. 1–5.

[12] A. Kushki, K. N. Plataniotis, and A. N. Venetsanopoulos, "Sensor selection for mitigation of rss-based attacks in wireless local area network positioning," in *IEEE ICASSP'08*, Las Vegas, NV, USA, March 2008, pp. 2065–2068.

[13] T. Li, Y. Chen, R. Zhang, Y. Zhang, and T. Hedgpeth, "Secure crowd-sourced indoor positioning systems," in *IEEE INFOCOM'18*, Honolulu, HI, March 2018.

[14] M. Youssef and A. Agrawala, "The horus wlan location determination system," in *ACM MobiSys'05*, Seattle, WA, June 2005, pp. 205–218.

[15] S. H. Fang, C. C. Chuang, and C. Wang, "Attack-resistant wireless localization using an inclusive disjunction model," *IEEE Transactions on Communications*, vol. 60, no. 5, pp. 1209–1214, May 2012.

[16] J.Morales, D. Akopian, and S. Agaian, "Faulty measurements impact on wireless local area network positioning performance," *IET Radar, Sonar & Navigation*, vol. 9, no. 5, pp. 501–508(7), June 2015.

[17] C. Laoudias, M. P. Michaelides, and C. G. Panayiotou, "Fault detection and mitigation in wlan rss fingerprint-based positioning," *Journal of Location Based Services*, vol. 6, no. 2, pp. 101–116, 2012.

[18] Y. Zhang, W. Liu, Y. Fang, and D. Wu, "Secure localization and authentication in ultra-wideband sensor networks," *IEEE J. Sel. Areas Commun.*, vol. 24, no. 4, pp. 829–835, Apr. 2006.

[19] S. Capkun and J. P. Hubaux, "Secure positioning of wireless devices with application to sensor networks," in *IEEE INFOCOM'05*, Miami, FL, March 2005.

[20] D. Liu, P. Ning, and W. K. Du, "Attack-resistant location estimation in sensor networks," in *ACM IPSN'05*, Los Angeles, California, April 2005, pp. 24–27.

[21] "Alfa awus036nha." [Online]. Available: https://store.rokland.com/products/alfa-awus036nha-802-11n-wireless-n-usb-wi-fi-adapter-2-watt?variant=128976802